

日本国特許庁  
PATENT OFFICE  
JAPANESE GOVERNMENT

#3  
JCS68 U.S. PTO  
106606/60  
09/909901  
07/28/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日

Date of Application:

2000年 7月27日

出願番号

Application Number:

特願2000-227096

出願人

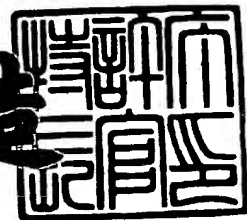
Applicant (s):

沖電気工業株式会社

2001年 3月30日

特許庁長官  
Commissioner,  
Patent Office

及川耕造



Inventors: Mihoko KITAMURA, et al  
ATTY DKT. 31759-173994

出証番号 出証特2001-3025138

【書類名】 特許願

【整理番号】 KN002337

【提出日】 平成12年 7月27日

【あて先】 特許庁長官 及川 耕造 殿

【国際特許分類】 G06F 17/27

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 村田 稔樹

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 北村 美穂子

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 下畑 さより

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 佐々木 美樹

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 福居 毅至

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会社  
社内

【氏名】 淵上 正睦

【特許出願人】

【識別番号】 000000295

【氏名又は名称】 沖電気工業株式会社

【代表者】 篠塚 勝正

【代理人】

【識別番号】 100090620

【弁理士】

【氏名又は名称】 工藤 宣幸

【手数料の表示】

【予納台帳番号】 013664

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9006358

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 自然言語処理装置、自然言語処理方法、自然言語パターン辞書作成装置及び自然言語パターン辞書作成方法

【特許請求の範囲】

【請求項 1】 言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置において、

予めパターン辞書に用意されている上記自然言語パターンの全て又は一部は、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有すると共に、

予めパターン辞書に用意されている上記自然言語パターンから、構文解析、及び又は、構文生成時の処理での候補として抽出された自然言語パターンが木構造に適合しているか否かを、素性による制約面からも検査するパターン検査手段と

適合する場合に、その自然言語パターンを木構造に適用すると共に、その自然言語パターンが中心要素情報を有するときに素性による制約を伝搬させるパターン適用手段と

を有することを特徴とする自然言語処理装置。

【請求項 2】 上記パターン適用手段は、素性情報を伝搬させる際に、予め非終端記号、終端記号毎に定まっている定義に従って、伝搬させる素性を制限することを特徴とする請求項 1 に記載の自然言語処理装置。

【請求項 3】 自然言語パターンにおける素性の制約として素性変数をも適用し、上記パターン検査手段及び上記パターン適用手段は、素性変数に対してのパターン適合検査やパターン適用を行なうことを特徴とする請求項 1 又は 2 に記載の自然言語処理装置。

【請求項 4】 パターン辞書に登録されている上記自然言語パターンは、その素性の制約情報を論理演算が実行し易い形式で保持するものであることを特徴とする請求項 1 ～ 3 のいずれかに記載の自然言語処理装置。

【請求項 5】 言語名、左辺、右辺の並びから構成された自然言語パターン

を利用して、構文解析、及び又は、構文生成を行う自然言語処理方法において、

予めパターン辞書に用意されている上記自然言語パターンの全て又は一部は、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有すると共に、

予めパターン辞書に用意されている上記自然言語パターンから、構文解析、及び又は、構文生成時の処理での候補として抽出された自然言語パターンが木構造に適合しているか否かを、素性による制約面からも検査するパターン検査工程と

適合する場合に、その自然言語パターンを木構造に適用すると共に、その自然言語パターンが中心要素情報を有するときに素性による制約を伝搬させるパターン適用工程と

を有することを特徴とする自然言語処理方法。

【請求項 6】 上記パターン適用工程は、素性情報を伝搬させる際に、予め非終端記号、終端記号毎に定まっている定義に従って、伝搬させる素性を制限することを特徴とする請求項 1 に記載の自然言語処理方法。

【請求項 7】 自然言語パターンにおける素性の制約として素性変数をも適用し、上記パターン検査工程及び上記パターン適用工程は、素性変数に対してのパターン適合検査やパターン適用を行なうことを特徴とする請求項 5 又は 6 に記載の自然言語処理方法。

【請求項 8】 パターン辞書に登録されている上記自然言語パターンは、その素性の制約情報を論理演算が実行し易い形式で保持するものであることを特徴とする請求項 5 ～ 7 のいずれかに記載の自然言語処理方法。

【請求項 9】 言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置に適用されるパターン辞書を作成する自然言語パターン辞書作成装置において、

左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有することがある、全てテキストデータで既述されている自然言語パターンを格納しているソース辞書と、

このソース辞書から読み出した自然言語パターンの素性制約情報を、論理演算

が実行し易い形式の素性制約データに変換して上記パターン辞書に格納する制約情報形式変換手段と

を有することを特徴とする自然言語パターン辞書作成装置。

【請求項 1 0】 上記制約情報形式変換手段は、

制約に用いる素性名及び素性値でなる素性情報の定義情報を格納している素性定義格納部と、

その定義情報に基づいて、論理演算が実行し易い形式のデータフォーマットを決定する素性制約データフォーマット決定部と、

決定されたデータフォーマットに従って、自然言語パターンの素性制約情報を、論理演算が実行し易い形式の素性制約データに変換する変換部とを有する

ことを特徴とする請求項 9 に記載の自然言語パターン辞書作成装置。

【請求項 1 1】 言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置に適用されるパターン辞書を作成する自然言語パターン辞書作成方法において、

左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有することがある、全てテキストデータで既述されている自然言語パターンを格納しているソース辞書から読み出した自然言語パターンの素性制約情報を、論理演算が実行し易い形式の素性制約データに変換して上記パターン辞書に格納する制約情報形式変換処理を有することを特徴とする自然言語パターン辞書作成方法。

【請求項 1 2】 上記制約情報形式変換処理は、

予め格納されている制約に用いる素性名及び素性値でなる素性情報の定義情報に基づいて、論理演算が実行し易い形式のデータフォーマットを決定する素性制約データフォーマット決定工程と、

決定されたデータフォーマットに従って、自然言語パターンの素性制約情報を、論理演算が実行し易い形式の素性制約データに変換する変換工程とを含む

ことを特徴とする請求項 1 1 に記載の自然言語パターン辞書作成方法。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、自然言語処理装置、自然言語処理方法、自然言語パターン辞書作成装置及び自然言語パターン辞書作成方法に関し、例えば、翻訳パターンを利用して機械翻訳を行う機械翻訳装置や機械翻訳方法などに適用し得るものである。

【0002】

【従来の技術】

従来型の機械翻訳装置において、システムやユーザが登録可能な翻訳知識（システム辞書やユーザ辞書）は、単語辞書、複数単語からなるイディオム、さらに、高性能なものでも「手を焼く」などの動詞と名詞の共起表現など、機械翻訳装置が予め決めている表現に限定されていた。これは、従来型の機械翻訳装置は、原言語の文法と辞書で原言語文を構文解析をし、変換辞書で言語変換し、目的言語の文法と辞書で目的言語文を生成するため、システムの文法に則った形式でしか、システムやユーザは翻訳辞書を登録できないためである。

【0003】

さらに、原言語と目的言語は、個別に解析、生成されるため、翻訳知識としてユーザの直感に即した

” [名詞句] を次に示します。”

” F o l l o w i n g i s [名詞句]. ” (名詞句は変数) (P1)  
 のような原言語と目的言語の対から構成されるパターンに基づいた辞書も登録できない。

【0004】

このような課題を解決したものとして、特開平5-290082号公報に記載された翻訳パターンに基づく翻訳方法及び翻訳装置がある。この公報記載の発明は、翻訳知識を文脈自由文法の範囲内で表し、文法規則を原言語のパターンと目的言語のパターンとで対にしたものである。また、文法規則はトライ型の辞書に格納することにより、構文解析の速さを実用レベルにすることを可能にした。さらに、文法規則を原言語のパターンと目的言語のパターンとで対にしたことにより、同期導出が可能となり、言語変換、構文生成は簡単な処理だけで済むようになった。

## 【 0 0 0 5 】

これにより、ユーザの直感に即した上述したパターン P 1 のような原言語と目的言語の対から構成される翻訳パターンに基づいた辞書が登録できるようになり、文脈自由文法の範囲内なら任意の翻訳知識を登録でき、翻訳処理できるようになった。

## 【 0 0 0 6 】

## 【発明が解決しようとする課題】

しかしながら、上記公報の技術では、全ての翻訳知識を文脈自由文法で記述する必要がある。

## 【 0 0 0 7 】

文脈自由文法では各変数（非終端記号）に制約を与えることはできず、制約を付与したい場合には、制約 1 つ 1 つに非終端記号を割り当て、それを用いた文法を書き下す必要があり、その結果、文法の数が増大になってしまい、メンテナンスが非常に困難になる。

## 【 0 0 0 8 】

そのため、文法規則に係る自然言語パターンに制約を加えることができ、加えたとしても、辞書の大容量化を避けることができる、構文解析や構文生成等を自然言語パターンを利用して行う自然言語処理装置や自然言語処理方法が望まれており、また、それらの自然言語処理装置や自然言語処理方法に好適な自然言語パターン作成装置及び自然言語パターン作成方法も望まれている。

## 【 0 0 0 9 】

## 【課題を解決するための手段】

かかる課題を解決するため、第 1 の本発明は、言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置において、予めパターン辞書に用意されている上記自然言語パターンの全て又は一部は、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有すると共に、予めパターン辞書に用意されている上記自然言語パターンから、構文解析、及び又は、構文生成時の処理での候補として抽出された自然言語パターンが木構造に適合してい



るか否かを、素性による制約面からも検査するパターン検査手段と、適合する場合に、その自然言語パターンを木構造に適用すると共に、その自然言語パターンが中心要素情報を有するときに素性による制約を伝搬させるパターン適用手段とを有することを特徴とする。

## 【 0 0 1 0 】

また、第 2 の本発明は、言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理方法において、予めパターン辞書に用意されている上記自然言語パターンの全て又は一部は、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有すると共に、予めパターン辞書に用意されている上記自然言語パターンから、構文解析、及び又は、構文生成時の処理での候補として抽出された自然言語パターンが木構造に適合しているか否かを、素性による制約面からも検査するパターン検査工程と、適合する場合に、その自然言語パターンを木構造に適用すると共に、その自然言語パターンが中心要素情報を有するときに素性による制約を伝搬させるパターン適用工程とを有することを特徴とする。

## 【 0 0 1 1 】

さらに、第 3 の本発明は、言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置に適用されるパターン辞書を作成する自然言語パターン辞書作成装置において、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有することがある、全てテキストデータで既述されている自然言語パターンを格納しているソース辞書と、このソース辞書から読み出した自然言語パターンの素性制約情報を、論理演算が実行し易い形式の素性制約データに変換して上記パターン辞書に格納する制約情報形式変換手段とを有することを特徴とする。

## 【 0 0 1 2 】

さらにまた、第 4 の本発明は、言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装

置に適用されるパターン辞書を作成する自然言語パターン辞書作成方法において、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有することがある、全てテキストデータで既述されている自然言語パターンを格納しているソース辞書から読み出した自然言語パターンの素性制約情報を、論理演算が実行し易い形式の素性制約データに変換して上記パターン辞書に格納する制約情報形式変換処理を有することを特徴とする。

#### 【 0 0 1 3 】

##### 【発明の実施形態】

##### （A）第 1 の実施形態

以下、本発明による自然言語処理装置、自然言語処理方法、自然言語パターン辞書作成装置及び自然言語パターン辞書作成方法を、機械翻訳装置及び機械翻訳方法に適用した第 1 の実施形態を図面を参照しながら詳述する。

#### 【 0 0 1 4 】

##### （A-1）第 1 の実施形態の構成

図 1 は、第 1 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。なお、実際上は、例えば、パソコンなどの情報処理装置上に、処理プログラムや固定データなどがローディングされて、第 1 の実施形態の機械翻訳装置が構築される。

#### 【 0 0 1 5 】

図 1 において、第 1 の実施形態の機械翻訳装置 1 は、入力部 1 0 1、出力部 1 0 2、形態素解析部 1 0 3、形態素生成部 1 0 4、構文解析部 1 0 5、構文生成部 1 0 6、パターン検査部 1 0 7、パターン適用部 1 0 8、パターン辞書 1 0 9、素性テーブル 1 1 0、辞書作成部 1 1 1、素性テーブル作成部 1 1 2、辞書ソース 1 1 3 及び素性定義表 1 1 4 からなる。

#### 【 0 0 1 6 】

なお、上記構成要素のうち、入力部 1 0 1、出力部 1 0 2、形態素解析部 1 0 3、形態素生成部 1 0 4、構文解析部 1 0 5、構文生成部 1 0 6、パターン検査部 1 0 7、パターン適用部 1 0 8 及びパターン辞書 1 0 9 が、機械翻訳装置本体を構成している。

## 【0017】

入力部101は、翻訳対象となる入力文（原言語文）を受け付けものであり、キーボードだけでなく、文章ファイルからの読み出し構成等も該当する。形態素解析部103は、その入力文は形態素（例えば単語）毎に区切るものである。なお、入力部101及び形態素解析部103は、従来のものと同様である。また、形態素解析に利用される形態素辞書の図示は省略している。

## 【0018】

構文解析部105は、形態素解析部103の形態素解析結果に基づき、パターン辞書109を辞書引きし、パターン検査部107及びパターン適用部108を適宜用いて、構文解析を行うものである。一方、構文生成部106は、構文解析部105から与えられた原言語文に対する構文解析結果に基づき、パターン辞書109を辞書引きし、パターン検査部107及びパターン適用部108を適宜用いて、目的言語での構文を生成するものである。構文解析部105や構文生成部106の機能については、後述する動作説明で明らかにする。

## 【0019】

形態素生成部104は、構文生成部106が生成した目的言語の構文に該当する形態素を当てはめて翻訳結果を得るものである。出力部102は、翻訳結果を出力するものであり、表示出力や印刷出力するものだけでなく、記憶出力するものも含まれる。また、出力する翻訳結果の数（候補数）をユーザが指定し得るものであっても良い。なお、形態素生成部104及び出力部102は、従来のものと同様である。また、形態素生成に利用される形態素辞書の図示は省略している。

## 【0020】

パターン辞書部109は、構文解析、構文生成に用いられる翻訳パターン（原言語パターン及び目的言語パターンの対）を格納しているものである。この第1の実施形態の場合、原言語パターン及び目的言語パターンのそれぞれには、後述するように、索性による制約が付加されていることがある。

## 【0021】

パターン検査部107は、構文解析結果の構文木又は生成しようとする構文木

が、辞書引きで得られたパターンの素性制約に適合しているかどうかを検査するものである。一方、パターン適用部 1 0 8 は、制約に適合している場合に、制約に適合しているパターンを適用した構文木に変更するものである。

#### 【 0 0 2 2 】

また、上述した構成要素のうち、素性テーブル 1 1 0、辞書作成部 1 1 1、素性テーブル作成部 1 1 2、辞書ソース 1 1 3 及び素性定義表 1 1 4 が、パターン辞書部 1 0 9 に格納させる翻訳パターンを作成する翻訳パターン辞書作成装置を構成している。これら各構成要素の機能については、動作の項で明らかにするが、簡単に言及すると、以下の通りである。

#### 【 0 0 2 3 】

素性定義表 1 1 4 は、制約などに利用し得る素性 (feature) を定義しているデータ (後述する図 5 参照) を記憶しているものである。素性テーブル 1 1 0 は、翻訳パターンに制約として付加し得る形式の制約素性データを記憶しているものである。素性テーブル作成部 1 1 2 は、素性定義表 1 1 4 に格納されている素性定義データから、素性テーブル 1 1 0 に格納する制約素性データを作成するものである。

#### 【 0 0 2 4 】

なお、素性テーブル 1 1 0、素性テーブル作成部 1 1 2 及び素性定義表 1 1 4 は、原言語及び目的言語別に機能するものである。

#### 【 0 0 2 5 】

辞書ソース 1 1 3 は、原翻訳パターンを格納しているものである。辞書作成部 1 1 1 は、辞書ソース 1 1 3 に格納されている原翻訳パターンを、適宜、素性テーブル 1 1 0 に格納されている制約素性データに基づいて、パターン検査部 1 0 7 やパターン適用部 1 0 8 が処理し易い形態の翻訳パターンのデータに変換してパターン辞書 1 0 9 に格納させるものである。

#### 【 0 0 2 6 】

(A-2) 第 1 の実施形態の動作

(A-2-1) 素性制約付き翻訳パターン辞書作成動作

素性による制約が付加された翻訳パターンの作成動作を説明する前に、まず、

翻訳パターンについて説明する。

【0027】

翻訳パターンは、図2に示すように、複数の言語によるパターンを対（組）にしているものである。なお、2言語間の片方向又は双方向の機械翻訳装置であれば、翻訳パターンは、2言語によるパターンを対にしているものである。各言語でのパターンは、文脈自由文法に従っており、言語種類を特定する言語名と、構文カテゴリー（パターン名）を表す左辺と、その構文カテゴリーを構成する他の構文カテゴリーの並びを構成する右辺（パターン構成要素）とで記述されている。

【0028】

例えば、構文解析時には、右辺（の非終端記号や終端記号）がその直前までの構文解析結果や形態素解析結果にマッチすれば、左辺（の非終端記号）に還元される。また例えば、構文生成時には、左辺（の非終端記号）が右辺（の非終端記号や終端記号）に置換される。

【0029】

図3は、素性による制約が付加されていない翻訳パターンの一例を示したものである。なお、図3は、辞書ソース113での格納例をも表している。

【0030】

図3において、e n及びj aはそれぞれ英語及び日本語を表しており、:を挟んでこれに続いているSは左辺の非終端記号名（構文カテゴリー名）であり、この例では、文を表す非終端記号名である。また、NP（名詞句）、VP（動詞句）は右辺を構成する非終端記号名（構文カテゴリー名）であり、これら非終端記号名に付加されている1、2は同じ番号のついた非終端記号同士が両パターン間で対応していることを示している。右辺を構成する各構文カテゴリー（パターン構成要素）は、大括弧の対によって区別されている。

【0031】

図4は、素性による制約が付加された翻訳パターンの一例を示したものである。なお、図4の例は、英語のパターンにのみ制約が付加され、日本語のパターンには制約が付加されていない例である。また、図4も、辞書ソース113での格

納例をも表している。

#### 【0032】

ここで、翻訳パターンに制約として付加される素性とは、形態素情報、構文情報、意味情報など翻訳で用いられる各種情報における特徴情報をいう。

#### 【0033】

図4において、numやposは素性名であり、前者は数（数量）を表す素性名であり、後者は品詞を表す素性名である。素性名に続く=s gや=nは素性値を表しており、前者は数量が単数であることを表しており、後者は品詞が名詞であることを表している。また、図4におけるVP：\*の\*は、この記号がついた右辺の要素（構文カテゴリー）が、右辺での中心要素であることを示している。構文解析時には、中心要素が持っている素性は全て左辺に対応した構文木のノードにコピーされる。コピーされた後、左辺にかかれた素性（この例ではtype=normal）自体もセットされる。

#### 【0034】

次に、パターン辞書109に格納する、制約素性データ付き翻訳パターンを作成するまでの動作を説明する。

#### 【0035】

なお、パターン辞書109に格納される翻訳パターンは、図3や図4に示すような翻訳パターン本体に、後述するパターン検査での処理を容易化するための制約素性データ（例えば、ビット配列データ）が付加されたものである。

#### 【0036】

まず、素性テーブル作成部112の動作を、図5のフローチャートを参照しながら説明する。

#### 【0037】

素性テーブル作成部112は、素性定義表114の格納内容（素性定義）を読み込む（ステップ501）。

#### 【0038】

図6は、素性定義表114での素性の定義データ例を示している。図6は、1個の素性定義対象に対する例を示している。図6において、／\*\*／で挟まれた

部分はデータ上のコメントであり、2行目は文字列 (string) 形式である素性で素性が標準形 baseForm として規定されていることを表しており、4行目は単語 (word) 形式である素性で素性が品詞 pos の素性値が n (名詞) 及び adj (形容詞) であることを定義しており、6行目は単語 (word) 形式である素性で素性が数 (数量) num の素性値が sg (単数) か pl (複数) であることを定義している。

## 【 0 0 3 9 】

以上のように、素性の定義は2つの種類があり、第1は、文字列 (文字列値) を持つものであり、第2は、予め定めた値を持つものである。

## 【 0 0 4 0 】

なお、素性定義は、図6に示すような既にデータ化されている既存の素性定義のものであって良い。

## 【 0 0 4 1 】

素性テーブル作成部 1 1 2 は、素性定義表 1 1 4 から素性定義を読み込むと、次に、素性定義で予め定めた値を持つとされたものに対し、図7に示すように、その素性と素性値の組毎に一意の番号 (組を規定する識別番号 N 1 ~ N 4) を付与し (ステップ 5 0 2)、その素性及び素性値の組と、その組に付与された番号の対応表 (以下、素性値 / 番号対応表やビット配列データと呼ぶ) を素性テーブル 1 1 0 に書き込む (ステップ 5 0 3)。

## 【 0 0 4 2 】

このような番号は、パターン検査部 1 0 7 が、後述するように、パターンの制約を検査するとき用いられ、これにより、パターン検査部 1 0 7 が高速に検査し得るようになされている。

## 【 0 0 4 3 】

図7は、品詞名素性 pos の素性値 n (名詞) に番号 N 1 を付与し、品詞名素性 pos の素性値 adj (形容詞) に番号 N 2 を付与し、数素性 num の素性値 sg に番号 N 3 を付与し、数素性 num の素性値 pl に番号 N 4 を付与した例である。図7は、あくまでも例であり、素性種類や各素性種類の素性値数もこれに限定されるものではない。

## 【0044】

パターンに対する素性による制約は、図8に示すように、4種類を基本としている。

## 【0045】

第1は、ある素性（素性名）に対して記述されている素性値で制約を与えるものである。上述した図4の英語用パターンにおける右辺第2項の  $num = sg$  は、この第1種類の制約例であり、数素性が単数であるという制約である。第2は、ある素性（素性名）に対して記述されている素性値以外という制約を与えるものである。例えば、 $num \neq sg$  という制約は、数素性が単数ではないという制約である。第3は、ある素性（素性名）に対して記述されている複数の素性値で制約を与えるものである。例えば、 $pos = n | adj$  という制約は、品詞素性が名詞か形容詞であるという制約である。第4は、ある素性（素性名）に対して記述されている複数の素性値以外という制約を与えるものである。例えば、 $pos \neq n | adj$  という制約は、品詞素性が名詞でもなくかつ形容詞でもないという制約である。

## 【0046】

なお、1個の構文要素（構文カテゴリー）に対する素性による制約は、複数以上の素性に対して付与されていても良い。上述した図4の英語用パターンにおける右辺第1項の  $num = sg : pos = n$  は、このような複数以上の素性制約の例であって、右辺第1項に係る構文要素（構文カテゴリー）に対し、数素性が単数、かつ、品詞素性が名詞という制約を付与したものである。上述のように、：によって、素性の違いを表している。

## 【0047】

辞書ソース113は、翻訳パターンが記述されているものであり、素性による制約が付与されている翻訳パターン（図4参照）も素性による制約が付与されていない翻訳パターン（図3参照）も含まれている。

## 【0048】

次に、辞書作成部111が辞書ソース113の格納内容（翻訳パターン本体）から、素性テーブル110の格納内容を参照して、パターン辞書109に登録す



る翻訳パターン（制約素性データ付き翻訳パターン）を作成する動作を、図9のフローチャートを参照しながら説明する。

#### 【0049】

辞書作成部111はまず、辞書ソース113の格納内容（翻訳パターン）を読み込む（ステップ901）。次に、辞書作成部111は、素性テーブル110の素性値／番号対応表をみながら、素性による制約を番号データ（ビット配列データ）に置き換えて（又は追加して）パターン辞書109に書き込む（ステップ902）。ここで、上述した図7に示した制約の種類や、制約が左辺に付与されているか右辺に付与されているかに応じて、ビット配列データが決定され、パターン辞書109に書き出される。

#### 【0050】

図10は、パターン（原言語パターン又は目的言語パターン）における素性による制約部分を、置き換えたビット配列データ（番号データ）の例を示すものである。

#### 【0051】

図10（A）は、右辺に存在する素性制約  $num = sg : pos = n$  に対するビット配列データを示している。この場合のビット配列データは、AND（論理積）値配列データと、ANDマスク配列データとでなる。素性値／番号対応表が上述した図7に示すものであるので、AND値配列データは「1010」となり、一方、AND値配列データ上で意味があるビット部分を示すANDマスク配列データは「1010」となる。

#### 【0052】

図10（B）は、右辺に存在する素性制約  $num \neq sg : pos = n$  に対するビット配列データを示している。この場合のビット配列データも、AND値配列データと、ANDマスク配列データとでなる。素性値／番号対応表が上述した図7に示すものであるので、AND値配列データは「1000」となり、一方、AND値配列データ上で意味があるビット部分を示すANDマスク配列データは「1010」となる。すなわち、ANDマスク配列データの3ビットが「1」であるので、AND値配列データでの3ビット目「0」は数素性が単数ではないこ

とを表している。

【0053】

図10(C)は、右辺に存在する素性制約  $pos = n | adj$  に対するビット配列データを示している。この場合のビット配列データは、OR（論理和）値配列データでなる。OR値配列データは「1100」となる。なお、OR値配列データに対するORマスク配列データは、OR値配列データ自体が有効部分の情報を含むので作成されない。

【0054】

図10(D)は、左辺に存在する素性制約  $num = sg$  に対するビット配列データを示している。この場合のビット配列データは、ADD値配列データと、ADDマスク配列データとでなる。左辺は、右辺の素性制約が伝搬され得ることもあるので（第2の実施形態参照）、ADD値配列データを採用し、この例の場合、ADD値配列データは「0010」となり、ADDマスク配列データは他の素性の伝搬を受け付けられるように「0011」となる。図10(D)に示すADD形式が適用されるのは、パターン適用部108の処理による（後述する図16参照）。

【0055】

素性の制約は、図8に示す表現が組み合わせられたとしても、図10に示す3種類の形式、すなわち、AND配列形式、OR配列形式及びADD配列形式のいずれかで表すことができる。

【0056】

なお、辞書作成部111によって、素性制約が置き換えられて作成されたビット配列データは、1や0が続けて並ぶ傾向にあるので（例えば、制約に係る素性種類が多い場合や素性値の種類が多い場合には）、メモリ空間効率や処理効率を上げるために圧縮していても良い。

【0057】

(A-2-2) 翻訳動作

次に、第1の実施形態の機械翻訳装置での翻訳パターンを利用した翻訳動作について説明する。

## 【0058】

入力部101が原言語文を取り込んで形態素解析部103に与え、その入力された原言語文を形態素解析部103が形態素解析する間での動作は、従来と同様である。

## 【0059】

以下、構文解析部105の動作を図11のフローチャートを参照しながら説明する。なお、この第1の実施形態の構文解析部105は、パターンに基づく従来の機械翻訳方法で述べられている処理に加えて、パターンに付加されている素性による制約に関する処理を行う。

## 【0060】

構文解析部105はまず、形態素解析部103から形態素解析済みの構文木を受け取る（ステップ1101）。そして、構文解析の必要な木がまだ残っていることを確認してから（ステップ1102）、パターン辞書109を引いて必要なパターン（ここでは原言語用のパターン）を取り出す（ステップ1103）。ここでは、複数のパターンが取り出されることがある。

## 【0061】

そして、適用するパターンが決まっていない状態で後述するステップ1105が実行されていない未処理のパターンが残っているか否かを判別する（ステップ1104）。残っていない場合には、上述したステップ1102に戻る。一方、未処理のパターンが残っていると、構文木とその未処理パターンをパターン検査部107に渡して制約に適合しているかどうかを検査させ（ステップ1105）、その検査結果を受け取って検査結果を判別する（ステップ1106）。

## 【0062】

なお、従来は素性による制約がないため取り出したパターンの検査は行なわれないが、この第1の実施形態では、パターンに制約を付加しているため、上述したステップ1105のパターン検査が実行される。

## 【0063】

適合していないという検査結果であれば、上述したステップ1104に戻り、これに対して、対象となっているパターンが制約に適合している場合には、その

構文木とパターンをパターン適用部 1 0 8 に渡して、その構文木にパターンを適用した結果の構文木を受け取り（ステップ 1 1 0 7）、上述したステップ 1 1 0 2 に戻る。

#### 【 0 0 6 4 】

ステップ 1 1 0 2 ～ 1 1 0 7 でなる処理ループを繰り返し実行し、構文解析が一通り終わったところで、構文解析結果の木を作成して構文生成部 1 0 6 に渡す（ステップ 1 1 0 8）。

#### 【 0 0 6 5 】

図 1 2 は、構文解析部 1 0 5 からの検査依頼時のパターン検査部 1 0 7 の処理を示すフローチャートである。パターン検査部 1 0 7 の図 1 2 で示す処理は、上述したステップ 1 1 0 5 での検査依頼に対応している。

#### 【 0 0 6 6 】

パターン検査部 1 0 7 はまず、構文木とパターンを受け取り（ステップ 1 2 0 1）、処理右辺数パラメータ  $i$  を初期値 0 に設定する（ステップ 1 2 0 2）。

#### 【 0 0 6 7 】

そして、パラメータ  $i$  がパターンの右辺数に達していないことを確認して（ステップ 1 2 0 3）、構文木の  $i + 1$  番目の子ノードと、パターンの右辺の  $i + 1$  番目を取り出してパターンの制約が満たされているか否かを判別し、その判別結果を認識する（ステップ 1 2 0 4、1 2 0 5）。なお、パターンの右辺の  $i + 1$  番目に制約がない場合には、制約が満たされていると取り扱う。

#### 【 0 0 6 8 】

構文木の  $i + 1$  番目の子ノードが、パターンの右辺の  $i + 1$  番目についての制約を満たしていない場合には、パターン検査部 1 0 7 は、適合しなかった旨を構文解析部 1 0 5 に返信する（ステップ 1 2 0 8）。

#### 【 0 0 6 9 】

これに対して、構文木の  $i + 1$  番目の子ノードが、パターンの右辺の  $i + 1$  番目についての制約を満たしている場合には、パラメータ  $i$  を 1 インクリメントしてステップ 1 2 0 3 に戻る（ステップ 1 2 0 6）。このステップ 1 2 0 3 でパラメータ  $i$  がパターンの右辺数に達したという判別結果を得る場合は、構文木の全

ての子ノードがパターンの対応する右辺要素の制約を満たした場合であり、このとき、パターン検査部 1 0 7 は、適合した旨を構文解析部 1 0 5 に返信する（ステップ 1 2 0 7）。

#### 【 0 0 7 0 】

ここで、素性の制約を調べるには通常かなりの時間や処理量やコストがかかるが、この第 1 の実施形態では、図 1 0 に示したように、ビット配列データで素性とその制約が保持されているので、ビット配列データを順に比較するだけで済み、時間や処理量やコストはかなり小さくなる。

#### 【 0 0 7 1 】

以下、ビット配列データを用いた制約適合性の判断方法の詳細を、図 1 3 及び図 1 4 に示す例を用いて説明する。

#### 【 0 0 7 2 】

図 1 3 は、素性による制約が、図 1 0 (A) や (B) に示すような AND 形式の場合を示している。符号 1 3 0 1 は、ある右辺要素に対するビット配列データ (AND 値配列データ及び AND マスク配列データ) を示している。符号 1 3 0 2 は、上記右辺要素に対応する構文木の子ノードのビット配列データを示している。このような子ノードのビット配列データは、この適合性の判断時に、子ノードの素性情報から形成される。なお、形態素解析部 1 0 3 が素性テーブル 1 1 0 の格納内容を参照して形成しておくようにしても良い。

#### 【 0 0 7 3 】

右辺要素に対するビット配列データと、子ノードのビット配列データとが照合されることになる (符号 1 3 0 3)。

#### 【 0 0 7 4 】

図 1 3 に示すような AND 形式の場合には、具体的には、子ノードの値配列データと AND マスク配列データのビット毎の AND をとり、子ノードの値配列データにおいて照合が必要なビット部分だけを浮き出させ、その結果が AND 値配列データと全く同じであれば制約に適合したと判断し、1 ビットでも異なっていれば制約に適合していないと判断する (符号 1 3 0 4)。

#### 【 0 0 7 5 】

図 1 3 の場合、構文木の子ノードのビット配列データ（符号 1 3 0 2）は、パターンの右辺要素に対するビット配列データ（符号 1 3 0 1）が表している制約に適合している場合である。

## 【 0 0 7 6 】

この図 1 3 から明らかなように、構文木の子ノードの値配列データは複数の素性に関するものであっても、パターンの右辺要素に対するビット配列データが表している制約に係る素性との適合だけが問題となる。

## 【 0 0 7 7 】

図 1 4 は、素性の制約が、図 1 0 (C) に示すような OR 形式の場合を示している。符号 1 4 0 1 は、パターンのある右辺要素に対するビット配列データ（OR 値配列データ）を示している。符号 1 4 0 2 は、上記右辺要素に対応する構文木の子ノードのビット配列データを示している。

## 【 0 0 7 8 】

OR 配列でも、右辺要素に対するビット配列データと、子ノードのビット配列データとが照合されることになる（符号 1 4 0 3）。

## 【 0 0 7 9 】

図 1 4 に示すような OR 形式の場合には、具体的には、子ノードの値配列データとパターンの OR 値配列データのビット毎の AND をとり、その結果に 1 ビットでも 1 があれば、制約に適合したと判断し、全てのビットが 0 であれば、制約に適合していないと判断する（符号 1 4 0 4）。

## 【 0 0 8 0 】

図 1 4 の場合、構文木の子ノードのビット配列データ（符号 1 4 0 2）は、パターンの右辺要素に対するビット配列データ（符号 1 4 0 1）が表している制約に適合している場合である。

## 【 0 0 8 1 】

この図 1 4 から明らかなように、OR 形式では、制約を付与する素性値が多くあり、構文木の子ノードの値配列データがそのうちの 1 個に関するものであっても制約を受けるようにすることを要する。

## 【 0 0 8 2 】

図 1 5 は、構文解析時でのパターン適用部 1 0 8 の処理を示すフローチャートである。パターン適用部 1 0 8 の処理は、上述したステップ 1 1 0 7（図 1 1 参照）の処理に対応している。なお、図 1 5 は、素性の伝搬面から記述している。

#### 【 0 0 8 3 】

パターン適用部 1 0 8 はまず、構文木とパターンを受け取る（ステップ 1 5 0 1）。パターンの中心ノードに対応する構文木の子ノードの素性を構文木のトップノードにセットする（ステップ 1 5 0 2）。これを素性の伝搬と呼ぶ。次に、パターンの左辺の素性情報を構文木のトップノードに追加する（ステップ 1 5 0 3）。

#### 【 0 0 8 4 】

なお、パターン適用部 1 0 8 の処理が実行されるのは、パターン検査部 1 0 7 での検査によって、構文木が制約を適合しているパターンについてであり、パターン適用部 1 0 8 の処理により、制約が適合している素性情報を、上位ノードに伝搬させている。

#### 【 0 0 8 5 】

図 1 6 は、パターン適用部 1 0 8 の上述した処理を、ビット配列データに対する演算処理で実行する方法を示している。

#### 【 0 0 8 6 】

図 1 6 において、符号 1 6 0 1 は、制約適合のパターンの左辺要素に対するビット配列データ（ADD 値配列データ及び ADD マスク配列データ）を示しており、符号 1 6 0 2 は、パターンの右辺中心要素に対応する構文木の子ノードのビット配列データ（値配列データ及びマスク配列データ）を示している。

#### 【 0 0 8 7 】

パターン適用部 1 0 8 はまず、符号 1 6 0 2 で示す子ノードのビット配列データ（素性）を構文木のトップノードにコピーする（符号 1 6 0 3）。次に、トップノードの素性（値配列データ及びマスク配列データ）に、パターンでの制約を反映させる。この処理は、次の（1）式及び（2）式で表すことができる（符号 1 6 0 4）。

#### 【 0 0 8 8 】

値配列 = ADD 値配列 | ( ~ ( ADD マスク配列 ) & 値配列 ) … ( 1 )

マスク配列 = ADD マスク配列 | マスク配列 … ( 2 )

なお、( 1 ) 式及び ( 2 ) 式において、| はビット毎の OR ( 論理和 ) を表し、~ はビット毎の NOT ( 論理否定 ) を表し、& はビット毎の AND ( 論理積 ) を表している。また、配列データにおけるデータの用語は省略している。

#### 【 0 0 8 9 】

( 1 ) 式は、パターンでの制約を受けた素性の素性値 ( ADD 値配列 ) と、その素性以外の素性についての子ノードから承継された素性値 ( ~ ( ADD マスク配列 ) & 値配列 ) とを合成したものを、トップノードでの素性にすることを表している。このような素性種類の増加に対応するように、値配列で有効なビットの位置を規定するマスク配列も、( 2 ) 式に示すように、2 種類のマスク配列を合成したものとなる。

#### 【 0 0 9 0 】

符号 1 6 0 5 は、制約適合のパターンの左辺要素に対するビット配列データが符号 1 6 0 1 で示す場合であり、かつ、パターンの右辺中心要素に対応する構文木の子ノードのビット配列データが符号 1 6 0 2 で示す場合であるときの処理後のトップノードでのビット配列データ ( 値配列データ及びマスク配列データ ; 素性 ) を示している。

#### 【 0 0 9 1 】

構文解析部 1 0 5 がパターン検査部 1 0 7 やパターン適用部 1 0 8 に適宜処理依頼しながら構文解析して得た構文解析結果は、構文生成部 1 0 6 に与えられ、構文生成部 1 0 6 は、目的言語での構文木情報を生成する。

#### 【 0 0 9 2 】

以下、構文生成部 1 0 6 の動作の詳細を図 1 7 のフローチャートを参照しながら説明する。なお、この第 1 の実施形態の構文生成部 1 0 6 は、パターンに基づく従来の機械翻訳方法で述べられている処理に加えて、パターンに付加されている素性による制約に関する処理も行う。

#### 【 0 0 9 3 】

なお、上述した図 3 及び図 4 では、目的言語 ( 日本語 ) については、素性の制



約が付されていない例であるが、当然に素性による制約が付与されていても良い。また、同じ原言語パターンに対する目的言語パターンが複数あっても良い。

【 0 0 9 4 】

構文生成部 1 0 6 はまず、構文解析部 1 0 5 から構文解析済みの構文木を受け取る（ステップ 1 7 0 1）。そして、構文生成が必要な木（木部分）がまだ残っていることを確認してから（ステップ 1 7 0 2）、パターン辞書 1 0 9 を引いて必要なパターン（ここでは目的言語パターン）を取り出す（ステップ 1 7 0 3）。ここでは、複数のパターンが取り出されることがある。

【 0 0 9 5 】

そして、適用するパターンが決まっていない状態で後述するステップ 1 7 0 5 が実行されていない未処理のパターンが残っているか否かを判別する（ステップ 1 7 0 4）。残っていない場合には、上述したステップ 1 7 0 2 に戻る。一方、未処理のパターンが残っていると、構文木とその未処理パターンをパターン検査部 1 0 7 に渡して制約に適合しているかどうかを検査させ（ステップ 1 7 0 5）、その検査結果を受け取って検査結果を判別する（ステップ 1 7 0 6）。

【 0 0 9 6 】

なお、従来は素性による制約がないため取り出したパターンの検査は行なわれないが、この第 1 の実施形態では、パターンに制約を付加しているため、上述したステップ 1 7 0 5 のパターン検査が実行される。この構文生成時のパターン検査の際には、トップダウンに構文生成を行うので、パターン検査部 1 0 7 は、フローチャートの図示は省略しているが、構文木のトップノードとパターンの右辺との素性適合を検査する。

【 0 0 9 7 】

適合していないという検査結果であれば、上述したステップ 1 7 0 4 に戻り、これに対して、対象となっているパターンが制約に適合している場合には、その構文木とパターンをパターン適用部 1 0 8 に渡して、その構文木にパターンを適用した結果の構文木を受け取り（ステップ 1 7 0 7）、上述したステップ 1 7 0 2 に戻る。

【 0 0 9 8 】

この際、パターン適用部 108 が適用させるのは、フローチャートの図示は省略しているが、構文木の各子ノードとパターンの右辺の各要素である（図 13、図 14 参照）。また構文木のトップノードの素性情報をパターンの右辺の中心要素に対応する構文木の子ノードにセットする。これは、トップノードから中心ノードへの素性の伝搬である。

#### 【0099】

ステップ 1702～1707 でなる処理ループを繰り返し実行し、構文生成が一通り終ったところで、構文生成結果の構文木を作成して形態素生成部 104 に渡す（ステップ 1708）。

#### 【0100】

形態素生成部 104 は、従来と同様にして、構文生成部 106 によって生成された構文木に対し、その素性等を反映しつつ、目的言語での形態素を当てはめて翻訳結果（訳文）を得、出力部 102 によって得られた訳文が出力される。

#### 【0101】

##### （A-3）第 1 の実施形態の効果

第 1 の実施形態の機械翻訳装置及び機械翻訳方法によれば、以下の効果を奏することができる。

#### 【0102】

a. 従来のパターンに基づく機械翻訳では、素性による制約と素性伝搬とがないため、種々の制約（意味による制約、数や性による制約、構文情報による制約など）をパターンに盛り込もうとすると、制約の相違毎に非終端記号や終端記号を用意してパターンに書き下さないといけなかった。そのため、パターン数が膨大になってしまう。

#### 【0103】

この第 1 の実施形態によれば、制約を非終端記号や終端記号に対する付加情報として簡潔な形でパターンに記述できるようになったため、パターンの数を減らすことができる。また、制約を非終端記号や終端記号に対する付加情報としてパターンに導入しているので、非終端記号や終端記号の種類を多くする必要がなく、また、付加情報も素性の一般的な表記を適用でき、その結果、パターンが理解

し易いものとなり、パターンのメインテナンスのコスト（処理時間や記憶容量等）を激減させることができる。

## 【 0 1 0 4 】

b. 従来は、制約処理がなかったために、他の翻訳手法を組み合わせなければならなかったが、制約処理を用いることでこまかな翻訳処理も含めて全てをパターンに基づく翻訳方法で翻訳できるようになった。そのため、本質的に学習可能な翻訳エンジンが作成できる。

## 【 0 1 0 5 】

c. パターンの制約情報を、ビット配列データにして辞書に登録しておくようにしたので、制約に適合しているか否かのパターン検査や、適用パターンに応じた構文木の修正等の処理を高速に実行することができる。

## 【 0 1 0 6 】

## (A-4) 第1の実施形態の変形実施形態

上記説明では、パターンの表現方法として図2のような形式を示し、また、素性定義表として図5のような形式を示したが、同じ情報量があるならば他の表現形式を用いても良い。

## 【 0 1 0 7 】

また、上記では、素性の表現に、照合や合成等のし易さを考慮してビット配列データを用いたが、照合や合成等の処理結果に同じ結果が得られるのならば、他の表現形式を用いても良い。

## 【 0 1 0 8 】

さらに、上記では、パターン辞書109に登録する内容を形成する構成要素110～114をも含むものを示したが、これら要素を別個の装置に持たせ、登録されたパターン辞書109を当該機械翻訳装置に移植するようにしても良い。

## 【 0 1 0 9 】

さらにまた、上記では、素性テーブル110はパターン辞書109と別に存在するように示したが、パターン辞書109内に素性テーブル110を格納するようにしても良い。

## 【 0 1 1 0 】

## (B) 第2の実施形態

次に、本発明による自然言語処理装置、自然言語処理方法、自然言語パターン辞書作成装置及び自然言語パターン辞書作成方法を、機械翻訳装置及び機械翻訳方法に適用した第2の実施形態を図面を参照しながら詳述する。

## 【0111】

## (B-1) 第2の実施形態の構成

図18は、第2の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

## 【0112】

図18において、第2の実施形態の機械翻訳装置2は、入力部1801、出力部1802、形態素解析部1803、形態素生成部1804、構文解析部1805、構文生成部1806、パターン検査部1807、パターン適用部1808、パターン辞書1809、素性テーブル1810、辞書作成部1811、素性テーブル作成部1812、辞書ソース1813及び素性定義表1814からなる。

## 【0113】

ここで、入力部1801、出力部1802、形態素解析部1803、形態素生成部1804、構文解析部1805、構文生成部1806、パターン検査部1807、パターン辞書1809、辞書作成部1811、辞書ソース1813は、第1の実施形態の対応要素と同一のものである。

## 【0114】

第2の実施形態の場合、素性定義表1814、素性テーブル作成部1812、素性テーブル1810及びパターン適用部1808が、第1の実施形態のものと異なっている。

## 【0115】

後述する動作の項で明らかにするが、素性定義表1814における素性定義が第1の実施形態と異なっており、それにより、素性テーブル作成部1812が作成した素性テーブル1810も第1の実施形態のものとは異なっている。また、この第2の実施形態の場合、パターン適用部1808が素性テーブル1810をも参照する点が異なっている。

【 0 1 1 6 】

( B - 2 ) 第 2 の実施形態の動作

次に、素性テーブル 1 8 1 0 に格納する内容を形成する際の動作について説明する。

【 0 1 1 7 】

素性定義表 1 8 1 4 は、上述した図 6 に示した定義部分に、図 1 9 に示す定義部分を加えたものを素性定義としている。なお、図 1 9 ではコメントを省略している。

【 0 1 1 8 】

図 1 9 に示した部分は、非終端記号及び終端記号（構文カテゴリー；構文木要素； c a t e g o r y ）の定義とそれを取り得る素性名の定義である。

【 0 1 1 9 】

\* の行は、全ての非終端記号及び終端記号が取り得る素性名の定義行であり、その素性名が標準形、出現形及び品詞を規定している。

【 0 1 2 0 】

" " の行は、終端記号が取り得る素性名であり、活用形を示している。

【 0 1 2 1 】

N P の行は、非終端記号名 N P （名詞句）が取り得る素性名を定義しており、その素性名は数素性及び意味素性である。V P の行は、非終端記号名 V P （動詞句）が取り得る素性名を定義しており、その素性名は数素性及び意味素性である。S の行は、非終端記号名 S （文）が取り得る素性名を定義しており、その素性名は数素性及び文型素性である。

【 0 1 2 2 】

以上のように、第 2 の実施形態では、非終端記号及び終端記号の種類によって、制約し得る素性名を定義している。

【 0 1 2 3 】

N P の行などは、素性の伝搬の際に、伝搬先の非終端記号や終端記号が取り得る素性のみ伝搬させることを意図した定義である。このように指定した素性のみが伝搬するようにすれば、不要な素性が伝搬せず、必要な情報のみが構文木に載

ることになり、理解しやすく、無駄もない。また、多数の構文解析候補が構文解析中に存在するときに、同じ素性を持つトップノードを持つ構文解析木はマージして構文解析のコストを大幅に削減することができるが、不要な素性が伝搬するためにマージできないことがよくある。素性の伝搬の制限は、この不要な素性のためにマージができないという問題をも解決することができ、大幅な処理効率の向上につながる。

## 【 0 1 2 4 】

素性テーブル作成部 1 8 1 2 は、図 1 9 の定義部分に対して、図 2 0 に示すような非終端記号や終端記号毎の素性マスク（ビット配列データ）を作成して、素性テーブル 1 8 1 0 に格納する。なお、素性テーブル作成部 1 8 1 2 は、第 1 の実施形態で説明した図 7 に示すような対応表（この第 2 の実施形態では図 2 0 に示下と同様な、ビット値を挿入できる箱）を作成して素性テーブル 1 8 1 0 に格納することも行っている。

## 【 0 1 2 5 】

図 2 0 に示すものは、各非終端記号毎と終端記号に対して、伝搬する素性マスクである。伝搬する際に素性データをこれでマスクすれば必要なもののみが伝搬されるしくみとなる。

## 【 0 1 2 6 】

図 2 0 において、NP の素性マスク（ビット配列データ）は、図 1 9 の素性定義から、NP における値をとる素性は品詞素性（図 1 9 の 1 行目による）、数素性（図 1 9 の 3 行目による）及び意味素性（図 1 9 の 3 行目による）であるので、これらを伝搬し得るようにしている。なお、意味素性の要素は 6 個としている。S の素性マスク（ビット配列データ）は、図 1 9 の素性定義から、S における値をとる素性は品詞素性、数素性及び文型素性であるので、これらを伝搬し得るようにしている。なお、文型素性の要素は 2 個としている。VP の素性マスクは、図示していないが、図示した場合には、NP と同様になる。

## 【 0 1 2 7 】

図 2 1 は、第 2 の実施形態の構文解析時でのパターン適用部 1 8 0 8 の処理を示すフローチャートである。パターン適用部 1 8 0 8 の処理は、上述したステッ

プ1107（図11参照）の処理に対応している。

#### 【0128】

パターン適用部1808は構文木とパターンを受け取り（ステップ2101）、パターンの中心ノードに対応する構文木の子ノードの素性を構文木のトップノードにセットする（ステップ2102）。但し、その際、素性テーブル1810からセット先の非終端記号や終端記号の素性マスク（図20）を取り出し、子ノードの素性をこの素性マスクでマスクしてから伝搬させる。同様に、パターンの左辺の素性情報を構文木のトップノードに追加する際も、素性マスクでマスクしてから追加する（ステップ2103）。

#### 【0129】

なお、ステップ2102及び2103のマスク処理は一度に行なっても構わない。また、素性マスクは、パターン適用部1808に内蔵させておくようにしても良い。

#### 【0130】

また、フローチャートの図示は省略しているが、構文生成時において、パターン適用部108が適用させるのは、構文木の各子ノードとパターンの右辺の各要素である。また構文木のトップノードの素性情報をパターンの右辺の中心要素に対応する構文木の子ノードにセットする。この際にも、図20に示すような素性マスクによって、非終端記号や終端記号毎に、その適用範囲を規定する。

#### 【0131】

##### （B-3）第2の実施形態の効果

第2の実施形態の機械翻訳装置及び機械翻訳方法によっても、第1の実施形態と同様な効果a～cを奏することができる。第2の実施形態によれば、さらに、以下の効果を奏することができる。

#### 【0132】

d. 指定した素性のみが伝搬するようにすれば、不要な素性が伝搬せず、必要な情報のみが構文木に載ることになり、理解しやすく、無駄もない。

#### 【0133】

e. 多数の構文解析候補が構文解析中に存在するときに、同じ素性を持つトッ

ブノードを持つ構文解析木はマージして構文解析のコストを大幅に削減することができるが、不要な素性が伝搬するためにマージできないことがよくある。素性の伝搬の制限は、この不要な素性のためにマージができないという問題をも解決することができ、大幅な処理効率の向上につながる。

【0134】

(B-4) 第2の実施形態の変形実施形態

辞書作成部1811が、素性テーブル1810にある素性マスクの情報を用いて、不適当な素性をパターンでの非終端記号や終端記号に付加されているか否かの確認を行い、不適当な素性が付加されているときに、それをエラーとして表示して、辞書作成者に訂正を促す用にしても良い。

【0135】

また、辞書作成部1811が、素性テーブル1810にある素性マスクの情報を用いて、未定義の非終端記号を用いていることを検出し、辞書作成者に訂正を促すようにしても良い。これによって、非終端記号のスペルミスなどを防ぐことができる。

【0136】

さらに、素性定義表として図19のような形式を示したが、同じ情報量があるなら他の表現形式を用いても良く、また、素性マスクの表現も、図20に示したビット配列データに限定されず、同じ処理結果が得られるのならば、他の表現形式を用いても良い。

【0137】

さらにまた、非終端記号や終端記号毎に定義された制約し得る素性名だけを伝搬制御する方法は、素性マスクを利用する方法以外の方法で行っても良い。

【0138】

(C) 第3の実施形態

次に、本発明による自然言語処理装置、自然言語処理方法、自然言語パターン辞書作成装置及び自然言語パターン辞書作成方法を、機械翻訳装置及び機械翻訳方法に適用した第3の実施形態を図面を参照しながら詳述する。

【0139】



## (C-1) 第3の実施形態の構成

図22は、第2の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

## 【0140】

図22において、第3の実施形態の機械翻訳装置3は、入力部2201、出力部2202、形態素解析部2203、形態素生成部2204、構文解析部2205、構文生成部2206、パターン検査部2207、パターン適用部2208、パターン辞書2209、素性テーブル2210、辞書作成部2211、素性テーブル作成部2212、辞書ソース2213、素性定義表2214及びコンテキストデータ格納部2215からなる。

## 【0141】

すなわち、構成要素として、パターン検査部2207とパターン適用部2208が用いるコンテキストデータ格納部2215が第2の実施形態に追加されている。また、第2の実施形態と比較した場合、その機能が多少変更されているものは、素性定義表2214、素性テーブル作成部2212、素性テーブル2210、パターン検査部2207及びパターン適用部2208であり、他の構成要素は、第2の実施形態の対応するものと同様である。

## 【0142】

## (C-2) 第3の実施形態の動作

この第3の実施形態の場合、翻訳パターンは、素性値に変数（以下、素性変数と呼ぶ）を含むこともあり得る。図23は、素性変数を含む翻訳パターンの例を示している。図23において、 $num = \{NUM\}$  は、素性名  $num$  の素性値として  $NUM$  という素性変数が適用されていることを表している。このような素性変数  $NUM$  を適用することにより、図23の例では、パターン上の右辺要素  $NP$  及び  $VP$  において、素性名  $num$  の素性値が共通値であるという制約を付与していることを意味する。なお、この素性変数部分には、例えば、構文解析時であれば、パターン適用部2208によるボトムアップの素性伝搬処理によって、具体的な素性値が挿入される。

## 【0143】

素性定義表 2214 には、図 24 のような定義が含まれていても良い。これは、第 1 の実施形態に係る図 6 に示す「数」の記述の追加であり、素性変数を利用できる素性名の定義である。素性変数を導入したことにより、同一素性でも区別したいことが多くなると推測され、そのため、同一素性に対して複数の素性名を付与している。すなわち、図 24 は、num と sNum と oNum が同じ値を取り、同じ素性変数でチェックできることを意味する。このように素性定義表で＝で結んでいない素性同士は同じ素性変数を用いることはできない。もちろん、同じ素性名同士は同じ素性変数を用いることができる。

## 【0144】

素性テーブル作成部 2212 は、このような素性変数用の定義を読み込むと、図 25 の下側に記述したようなビット配列データを素性テーブル 2210 に書き出す。ここで、素性変数にはいずれの値（sg、p1）が挿入されるかわからないので共に 1 にしておく。また、オフセット値は、その素性変数の本来のビット配列データ（例えば、第 1 の実施形態に係る図 7）での位置を示している。辞書作成部 2211 は、辞書ソース 2213 からの翻訳パターンに素性変数があれば、図 25 に示すような置き換えを行う。

## 【0145】

パターン検査部 2207 は、素性の制約を検査する際、素性変数が使われていれば、その素性値をコンテキストデータ格納部 2215 に格納する。また、格納する際、既に同じ変数名のコンテキストデータがあった場合、そのコンテキストデータ格納部 2215 にある変数値と AND（ビット論理積）をとり、その結果が全て 0 になったら、制約に不適合とする。1 ビットでも残れば、その結果をコンテキストデータ格納部 2215 に格納し直す。

## 【0146】

パターン適用部 2208 は、素性変数が使われていれば、コンテキストデータ格納部 2215 からその変数名のデータを取り出し、セットするノードの値配列の図 25 で示されたオフセットの場所に、コンテキストデータを図 25 の配列の大きさだけセットする。

## 【0147】

## (C-3) 第3の実施形態の効果

第3の実施形態の機械翻訳装置及び機械翻訳方法によっても、既述の実施形態と同様な効果a～eを奏することができる。第3の実施形態によれば、さらに、以下の効果を奏することができる。

## 【0148】

f. 右辺のあるノードとあるノードのある素性の値を同じにしたいとき、第3の実施形態がなかった場合、素性とり得る値の数だけパターンが必要だった。しかし、素性変数を用いると、1つのパターンで書くことができ、構文解析などの速度を速め、メンテナンスコストも大幅に下げることができる。

## 【0149】

## (C-4) 第3の実施形態の変形実施形態

翻訳パターンとして図23のような形式を示したが、同じ情報量があるなら他の表現形式を用いても良い。素性定義表として図24のような形式を示したが、同じ情報量があるなら他の表現形式を用いても良い。素性変数の表現に図25に示したビット配列データを用いたが、同じ結果が得られるのならば、他の表現形式を用いても良い。

## 【0150】

## (D) 他の実施形態

上記各実施形態は、翻訳パターンを利用した機械翻訳装置及び方法に本発明を適用したものを示したが、上記各実施形態は、構文解析処理や構文生成処理のそれぞれの処理に特徴があり、そのため、上記各実施形態の技術思想を他の自然言語処理装置や方法に適用することができる。例えば、質疑応答装置で回答文をパターン（自然言語パターン）を用いて構文解析するまでの部分に本発明の技術思想を適用することができる。また回答文に応じたさらなる質問文の構文木から構文生成処理する場合にも、本発明の技術思想を適用することができる。

## 【0151】

また、構文解析処理だけを含む自然言語処理装置及び方法や、構文生成処理だけを含む自然言語処理装置及び方法にも、本発明を適用することができる。

## 【0152】

【発明の効果】

本発明によれば、文法規則に係る自然言語パターンに制約を加えることができ、加えたとしても、辞書の大容量化を避けることができる、構文解析や構文生成等を自然言語パターンを利用して行う自然言語処理装置や自然言語処理方法を実現できる。

【0153】

また、本発明によれば、それらの自然言語処理装置や自然言語処理方法に好適な自然言語パターン作成装置及び自然言語パターン作成方法を実現できる。

【図面の簡単な説明】

【図1】

第1の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【図2】

翻訳パターンの原則的な構成例を示す説明図である。

【図3】

素性による制約が付加されていない翻訳パターンの一例を示す説明図である。

【図4】

第1の実施形態の素性による制約が付加されている翻訳パターンの一例を示す説明図である。

【図5】

第1の実施形態の素性テーブル作成部112の動作を示すフローチャートである。

【図6】

第1の実施形態の素性定義例を示す説明図である。

【図7】

第1の実施形態の素性名と素性値との組への番号付与例を示す説明図である。

【図8】

第1の実施形態の素性による制約種類例を示す説明図である。

【図9】

第1の実施形態の辞書作成部111の動作を示すフローチャートである。

【図 1 0】

第 1 の実施形態の素性による制約のビット配列データ例を示す説明図である。

【図 1 1】

第 1 の実施形態の構文解析部 1 0 5 の処理を示すフローチャートである。

【図 1 2】

第 1 の実施形態のパターン検査部 1 0 7 の処理を示すフローチャートである。

【図 1 3】

第 1 の実施形態のビット配列データを用いた制約適合性の判断方法の説明図（  
1）である。

【図 1 4】

第 1 の実施形態のビット配列データを用いた制約適合性の判断方法の説明図（  
2）である。

【図 1 5】

第 1 の実施形態のパターン適用部 1 0 8 の処理を示すフローチャートである。

【図 1 6】

第 1 の実施形態のパターン適用部 1 0 8 の具体的処理方法の説明図である。

【図 1 7】

第 1 の実施形態の構文生成部 1 0 6 の処理を示すフローチャートである。

【図 1 8】

第 2 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【図 1 9】

第 2 の実施形態の素性定義例を示す説明図である。

【図 2 0】

第 2 の実施形態の素性マスク例を示す説明図である。

【図 2 1】

第 2 の実施形態のパターン適用部 1 8 0 8 の処理を示すフローチャートである。

【図 2 2】

第 3 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【図 2 3】

第 3 の実施形態の素性変数を含む翻訳パターン例を示す説明図である。

【図 2 4】

第 3 の実施形態の素性変数に係る素性定義の説明図である。

【図 2 5】

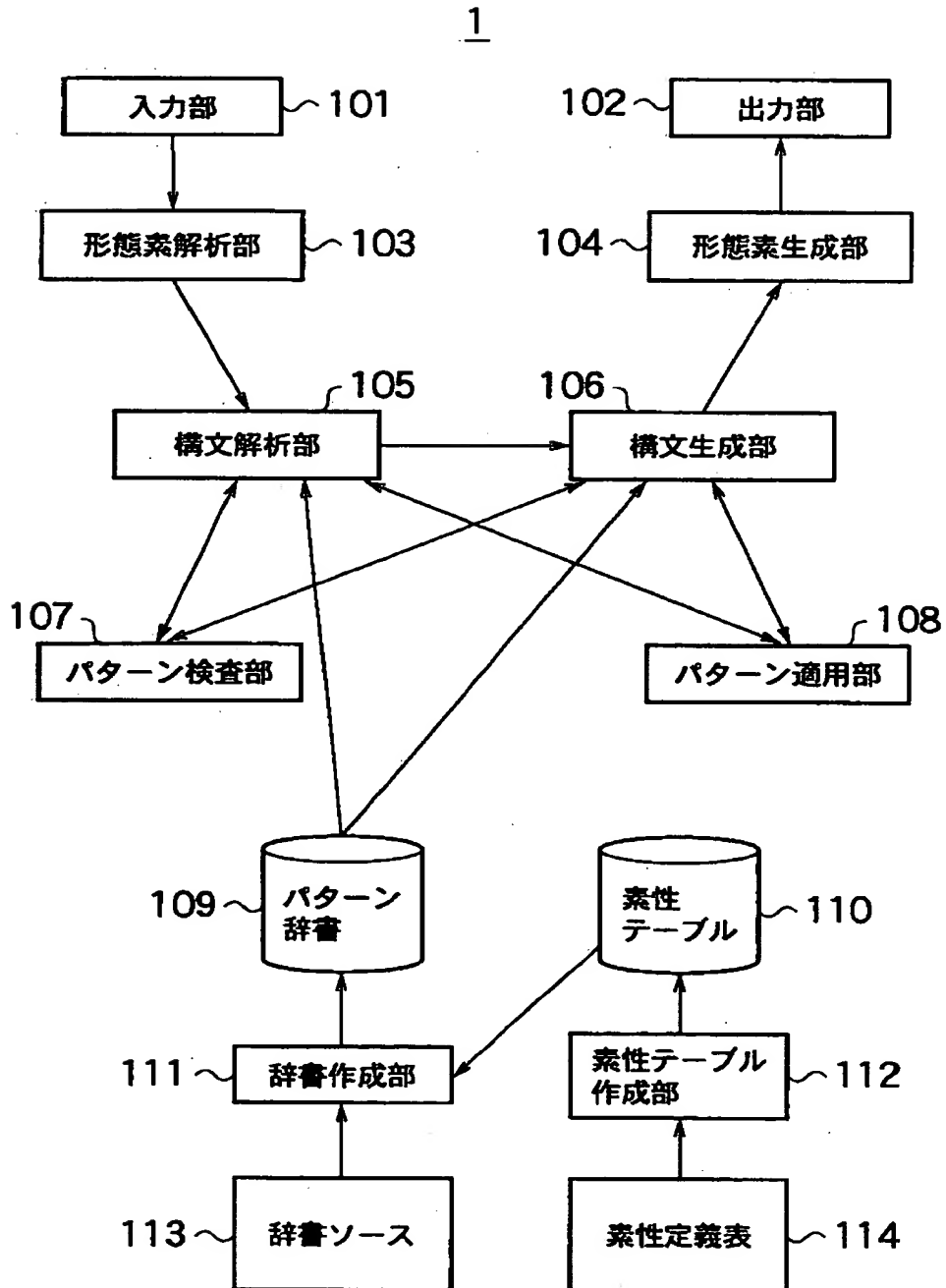
第 3 の実施形態の素性変数に係るビット配列データの説明図である。

【符号の説明】

- 1、2、3 …機械翻訳装置、
- 1 0 5、1 8 0 5、2 2 0 5 …構文解析部、
- 1 0 6、1 8 0 6、2 2 0 6 …構文生成部、
- 1 0 7、1 8 0 7、2 2 0 7 …パターン検査部、
- 1 0 8、1 8 0 8、2 2 0 8 …パターン適用部、
- 1 0 9、1 8 0 9、2 2 0 9 …パターン辞書、
- 1 1 0、1 8 1 0、2 2 1 0 …素性テーブル、
- 1 1 1、1 8 1 1、2 2 1 1 …辞書作成部、
- 1 1 2、1 8 1 2、2 2 1 2 …素性テーブル作成部、
- 1 1 3、1 8 1 3、2 2 1 3 …辞書ソース、
- 1 1 4、1 8 1 4、2 2 1 4 …素性定義表、
- 2 2 1 5 …コンテキストデータ格納部。

【書類名】 図面

【図1】



【図 2】

[ 言語名 1 : 左辺 1 右辺の並び 1 ]  
 [ 言語名 2 : 左辺 2 右辺の並び 2 ]  
 [ 言語名 3 : 左辺 3 右辺の並び 3 ]  
 ⋮  
 [ 言語名 n : 左辺 n 右辺の並び n ];

【図 3】

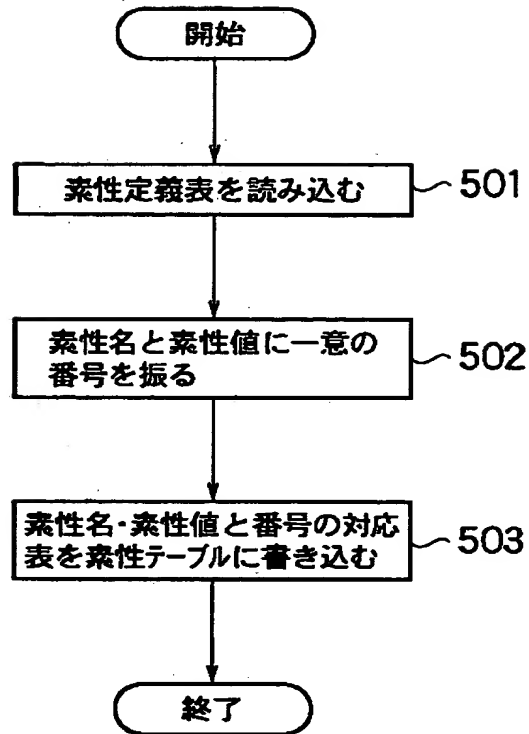
[ en : S [ 1 : NP ] [ 2 : VP ] ]  
 [ ja : S [ 1 : NP ] は [ 2 : VP ] ];

【図 4】

[ en : S : type = normal [ 1 : NP : num = sg : pos = n ] [ 2 : VP : \* : num = sg ] ]  
 [ ja : S [ 1 : NP ] は [ 2 : VP ] ];



【図 5】

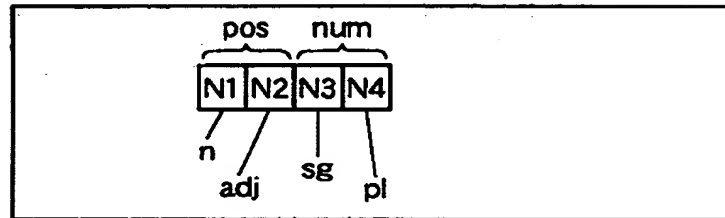


【図 6】

```

/* 標準形 */
string          baseForm ;
/* 品詞名 */
word            pos = { n   adj   } ;
/* 数 */
word            num = { sg   pl   } ;
    
```

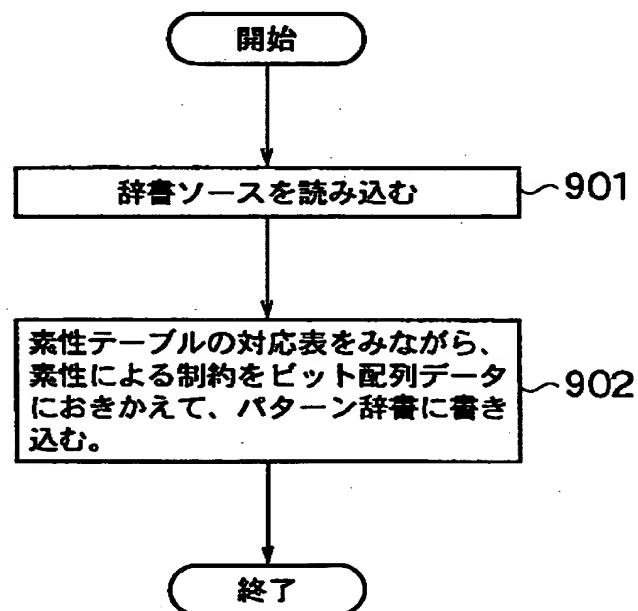
【図 7】



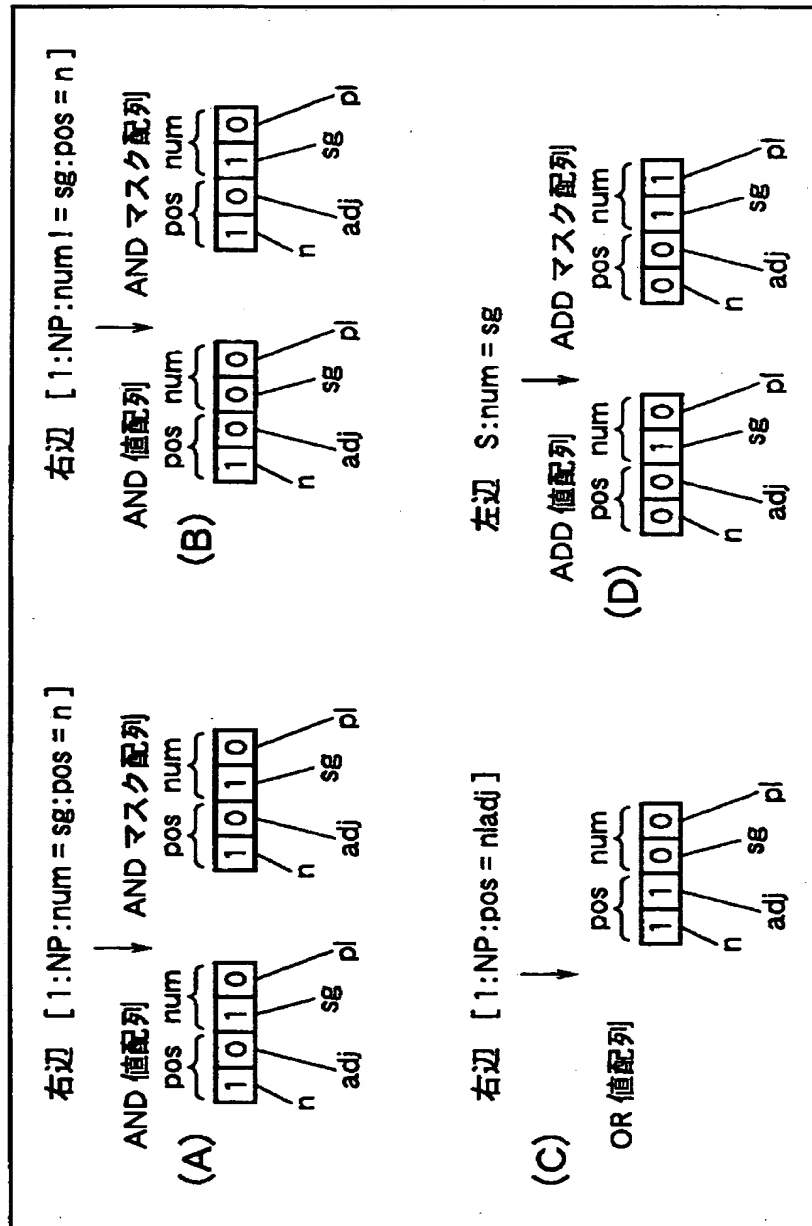
【図 8】

素性名 = 素性値  
 素性名 != 素性値  
 素性名 = 素性値 1 | 素性値 2 | ... | 素性値 x  
 素性名 != 素性値 1 | 素性値 2 | ... | 素性値 x

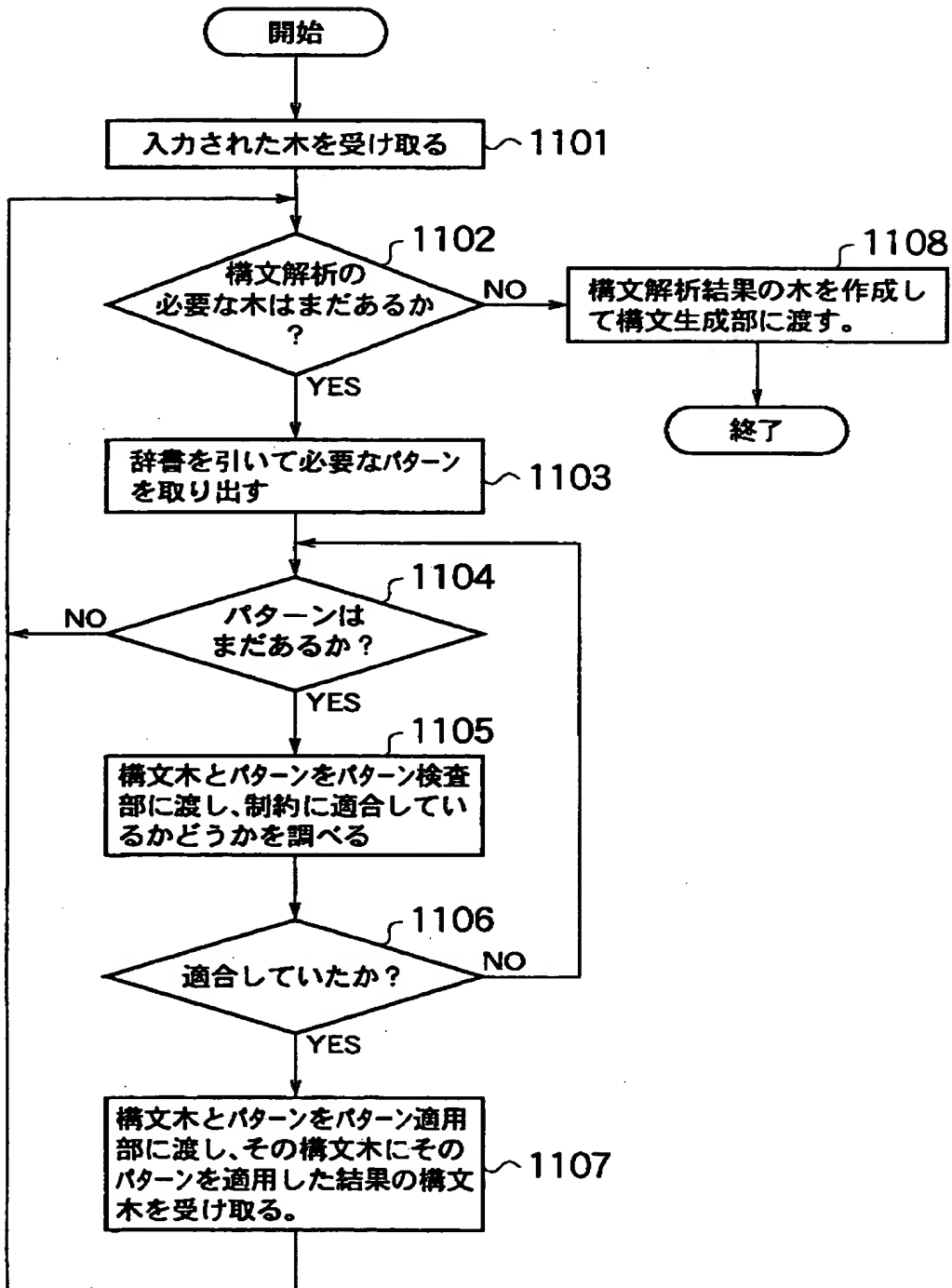
【図 9】



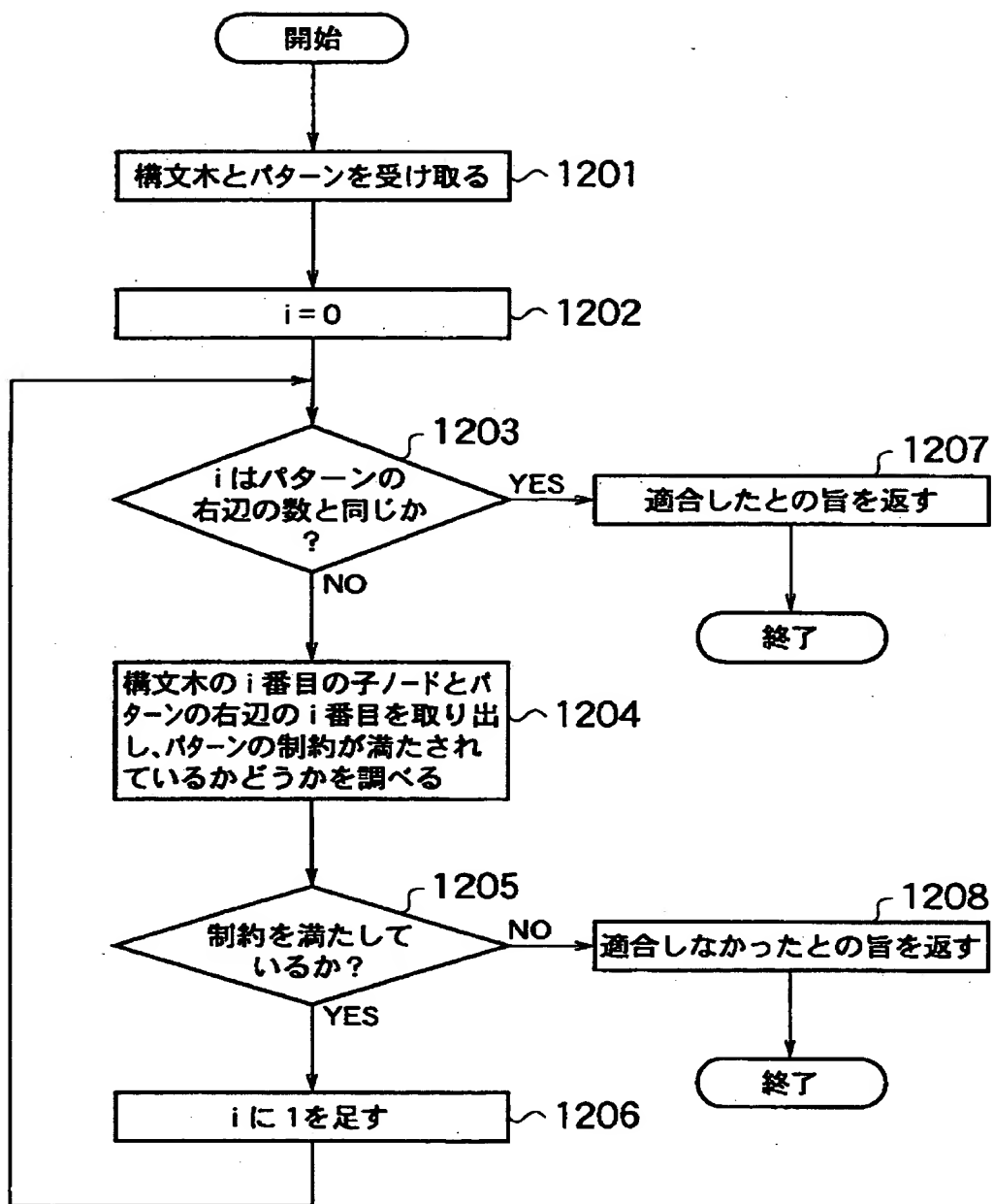
【図 10】



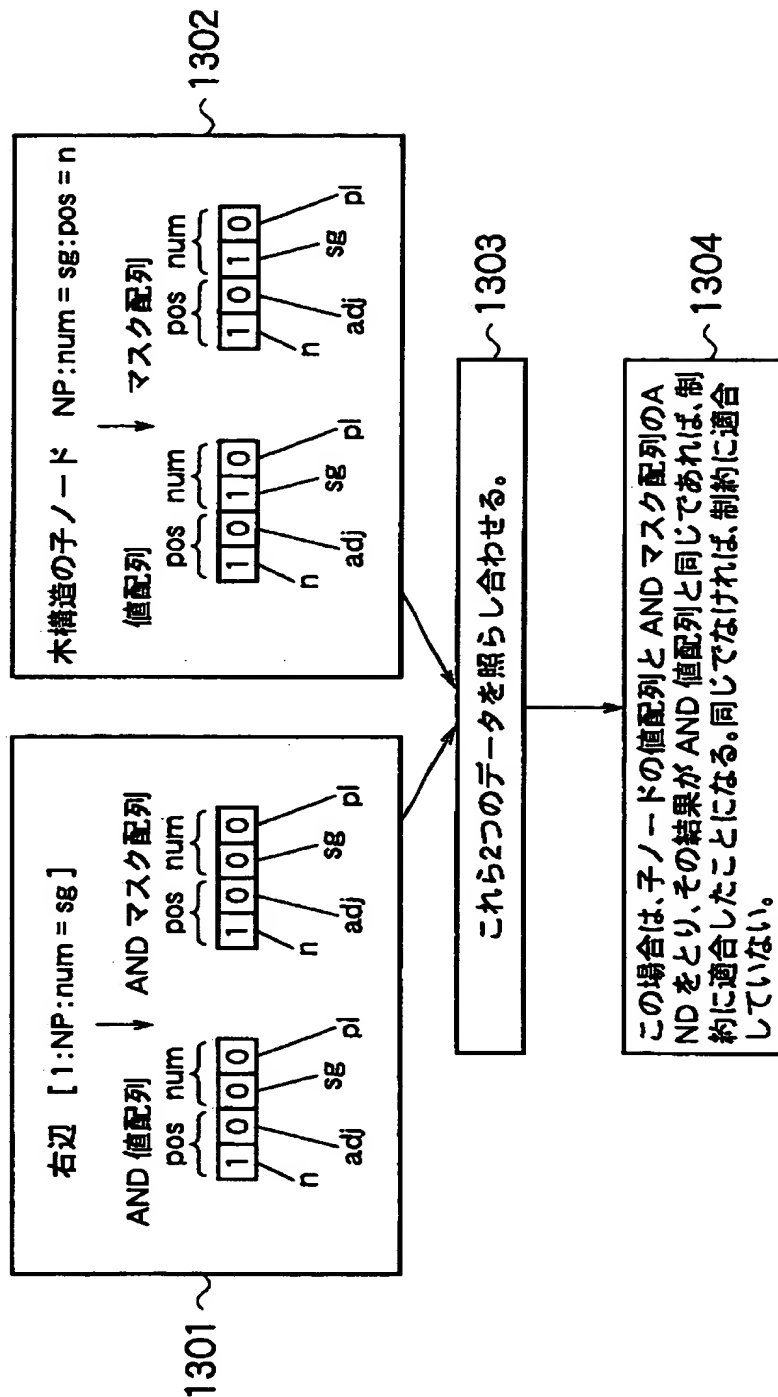
【図 11】



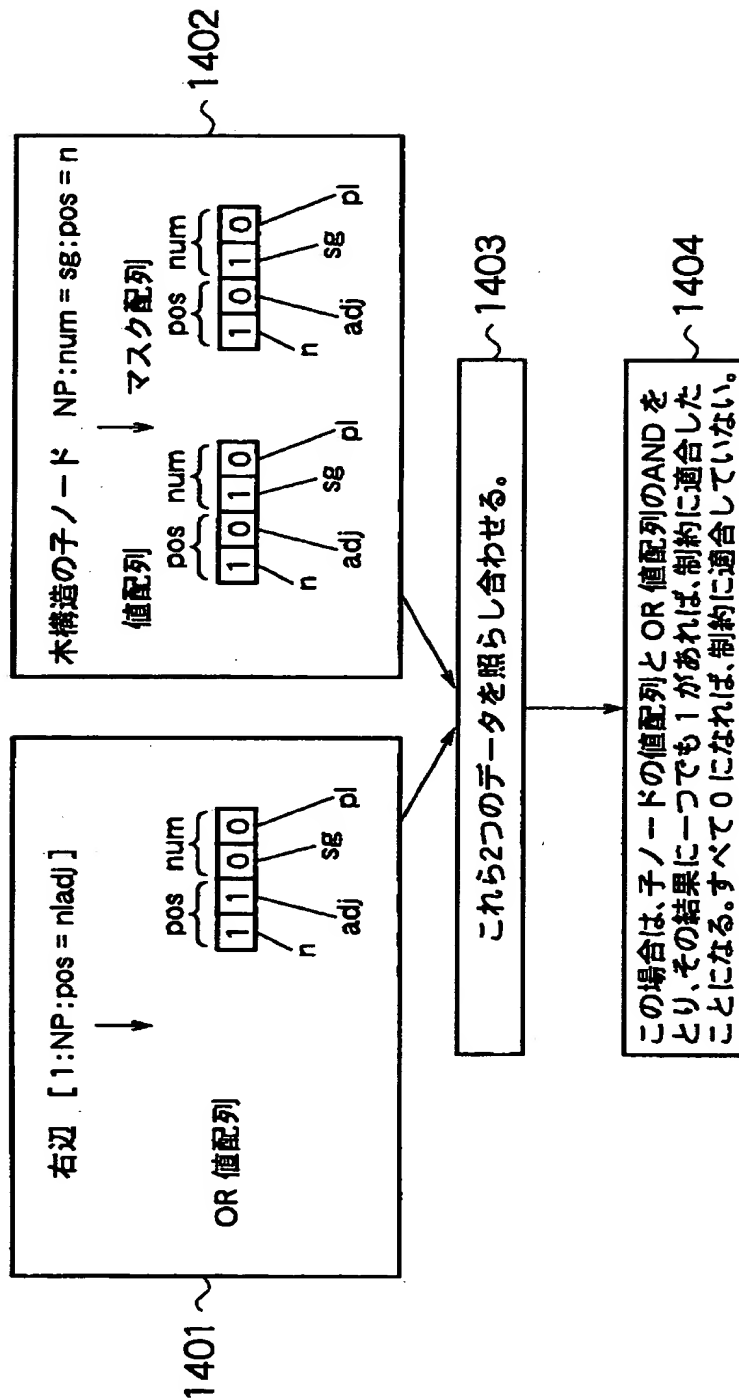
【図 12】



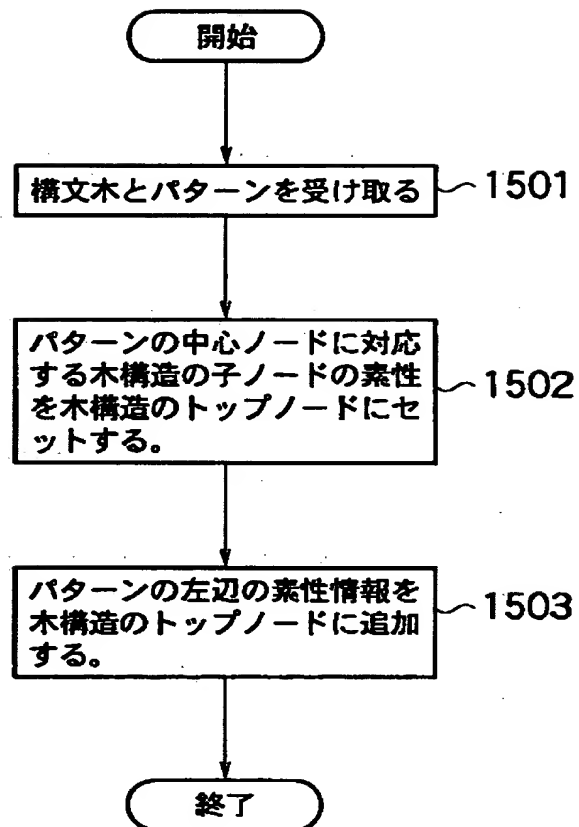
【図 13】



【図14】

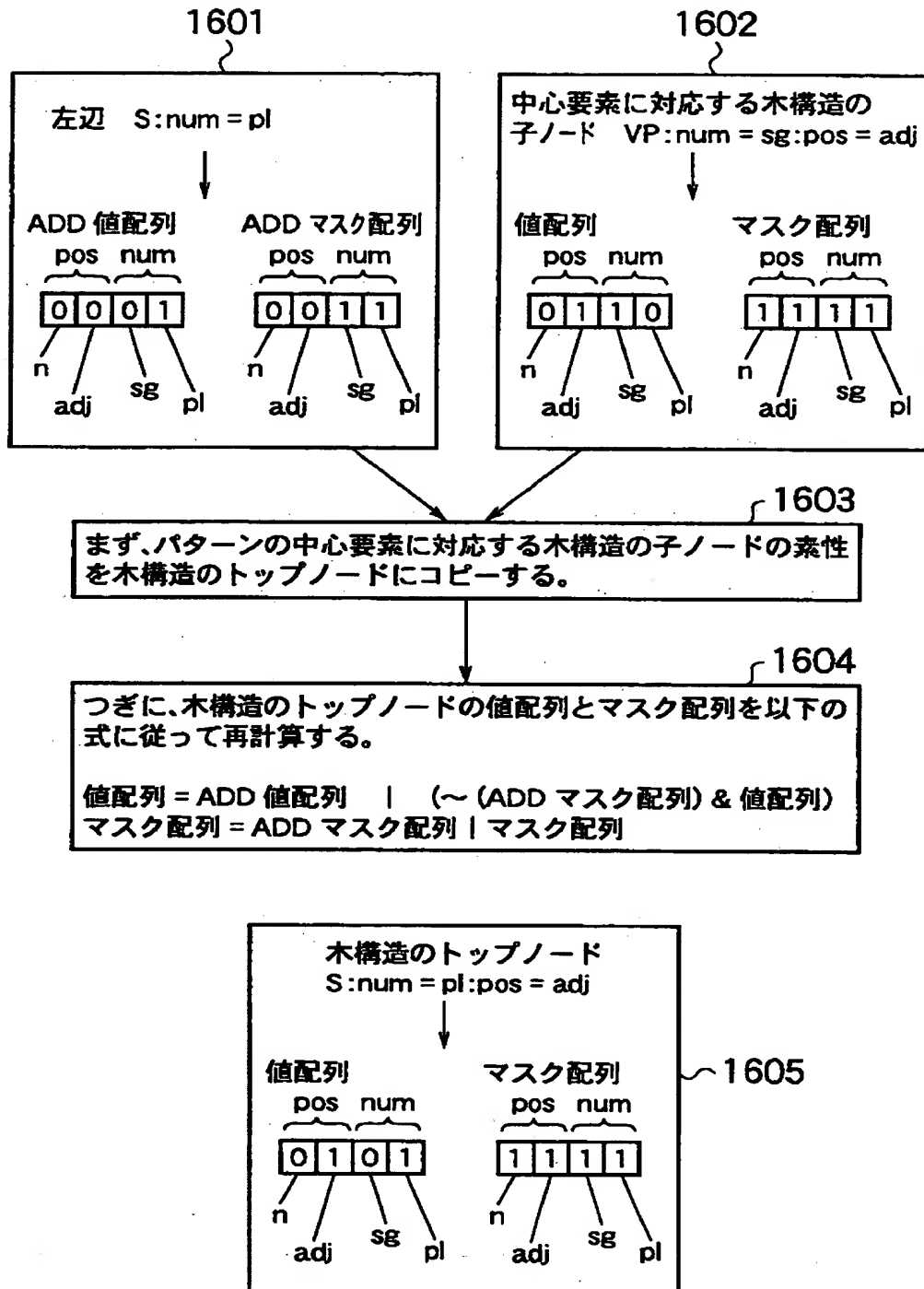


【図 1 5】

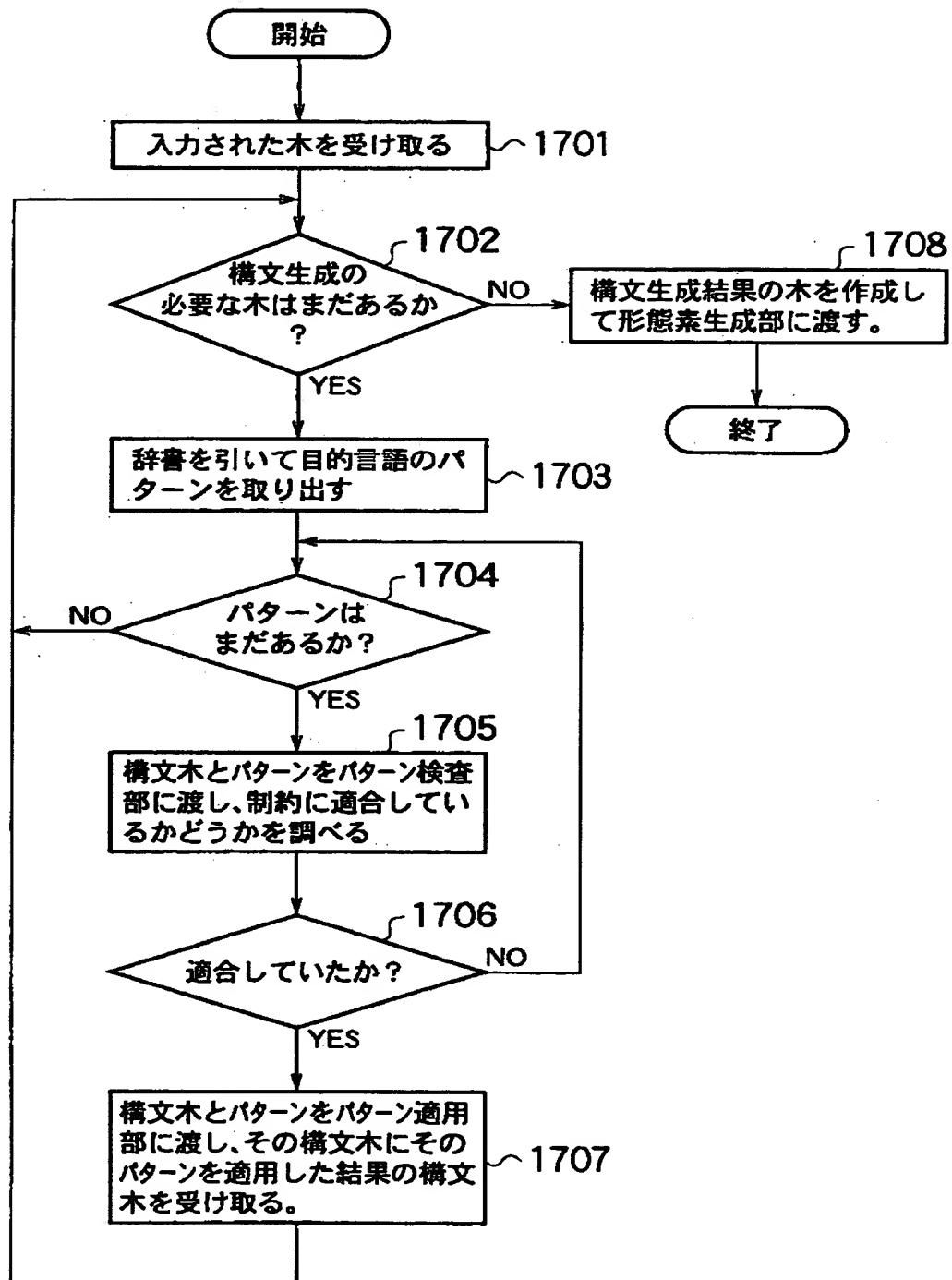




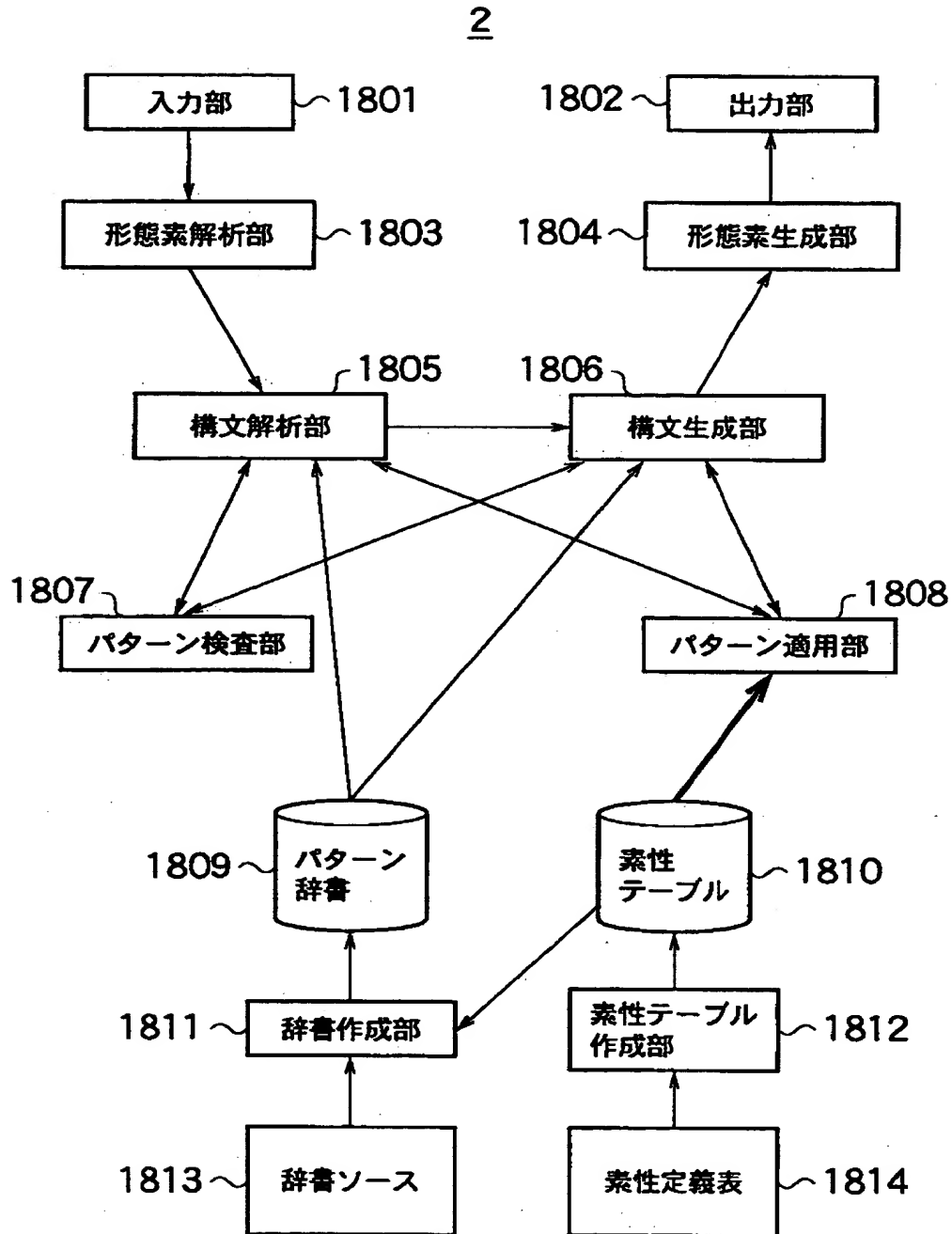
【図 16】



【図 17】



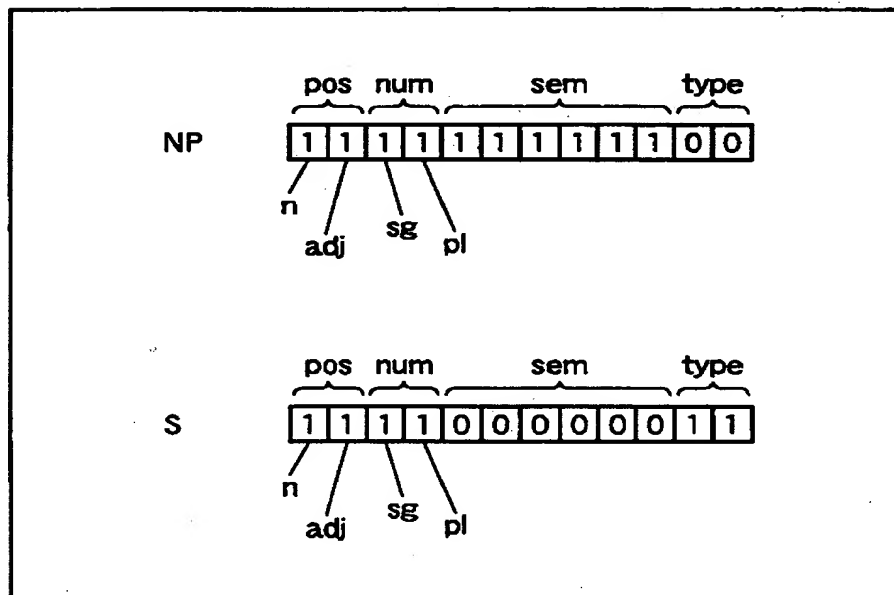
【図18】



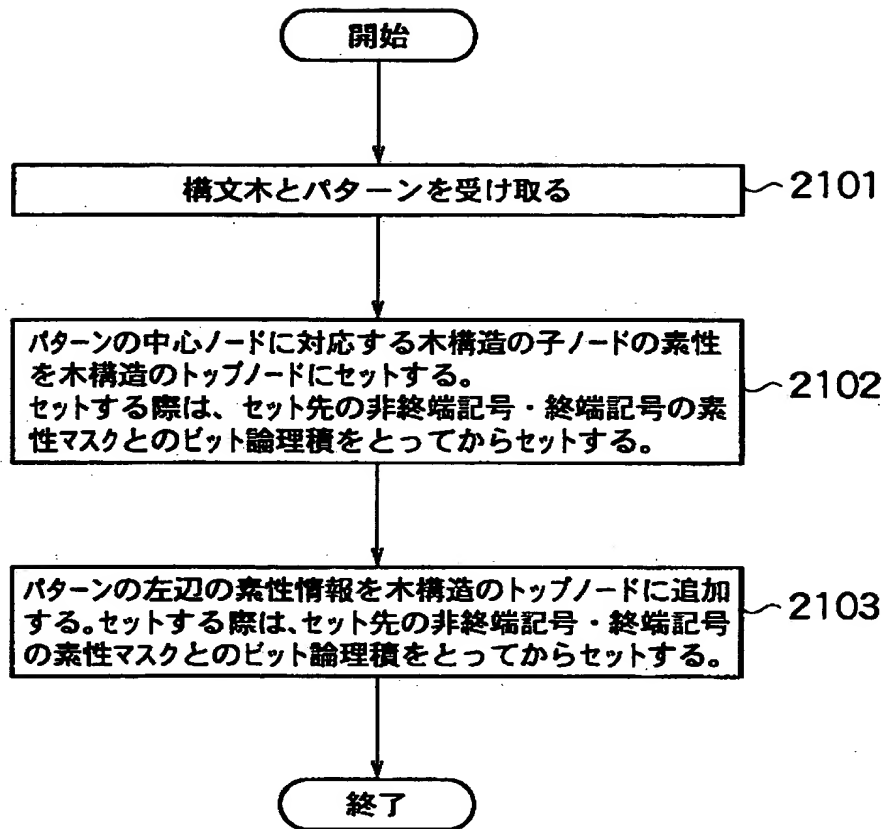
【図19】

category	*={ baseForm surfaceForm pos };
category	""={ inf };
category	NP={ num sem };
category	VP={ num sem };
category	S={ num type };

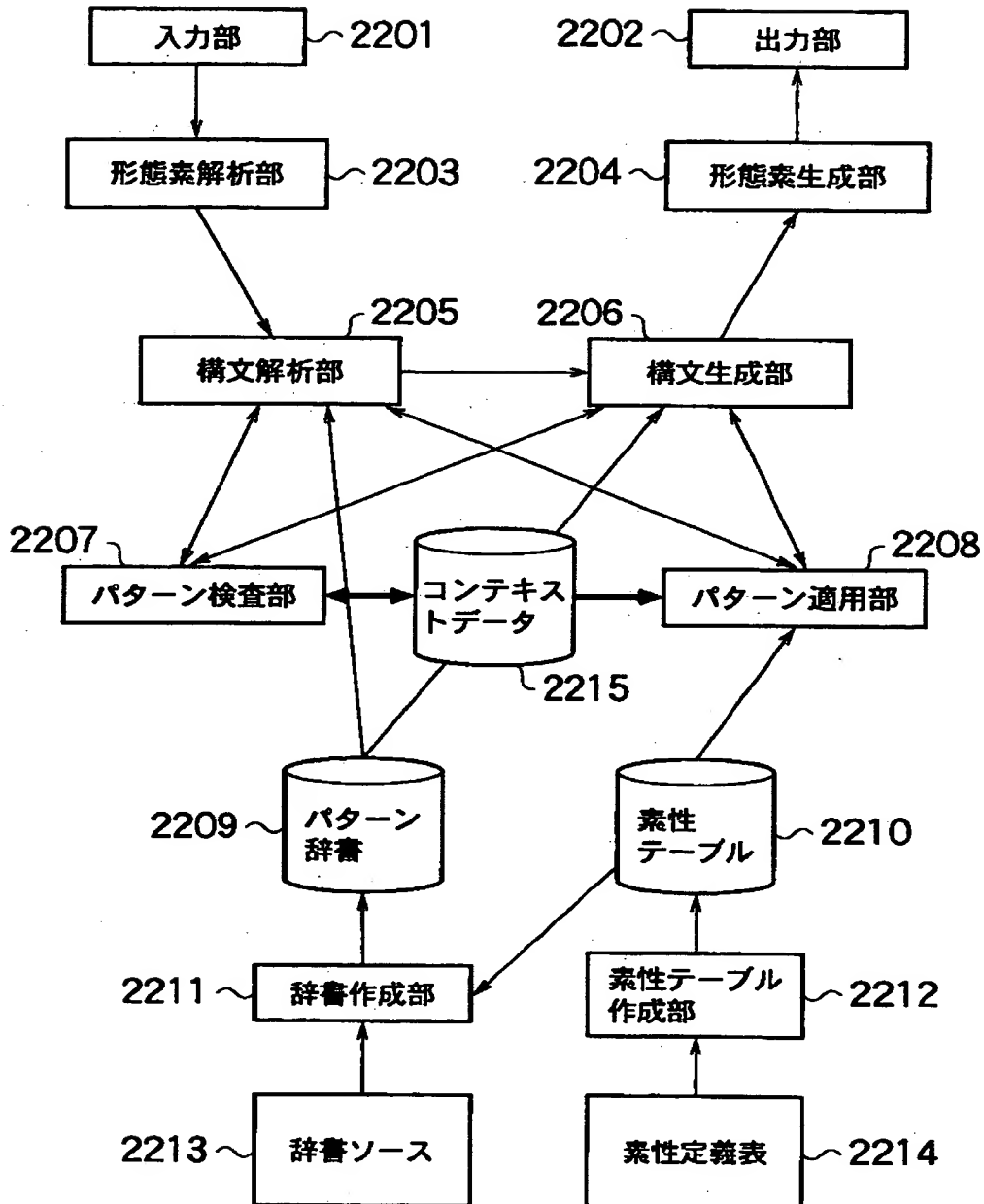
【図20】



【図 21】



【図 22】



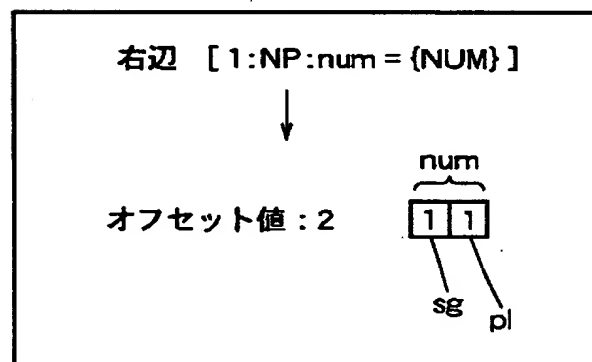
【図 2 3】

```
[ en:S:type = normal [ 1:NP:num = {NUM};pos = n] [ 2:VP:*.num = {NUM}]]
[ ja:S                [ 1:NP ]は [ 2:VP ]];
```

【図 2 4】

```
/* 数 */
word num = sNum = oNum = {sg pl};
```

【図 2 5】



【書類名】 要約書

【要約】

【課題】 自然言語パターンに制約を加えることができ、加えたとしても、辞書の大容量化を避けることができる自然言語処理装置及び方法を実現する。

【解決手段】 本発明は、言語名、左辺、右辺の並びから構成された自然言語パターンを利用して、構文解析や構文生成を行う自然言語処理装置及び方法に関する。そして、自然言語パターンの全て又は一部が、左辺及び又は右辺に、素性による制約、及び、素性伝搬時の中心のパターン要素を規定する中心要素情報を有すると共に、抽出された自然言語パターンが木構造に適合しているか否かを、素性による制約面からも検査するパターン検査手段（工程）と、適合する場合に、その自然言語パターンを木構造に適用すると共に、その自然言語パターンが中心要素情報を有するときに素性による制約を伝搬させるパターン適用手段（工程）とを有することを特徴とする。

【選択図】 図 1

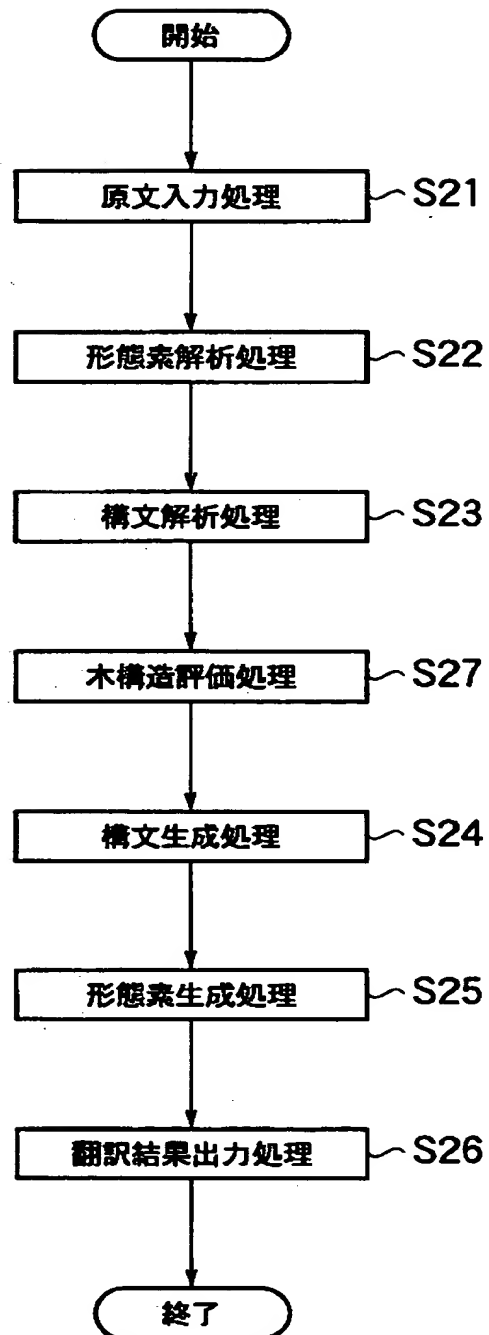


出 願 人 履 歴 情 報

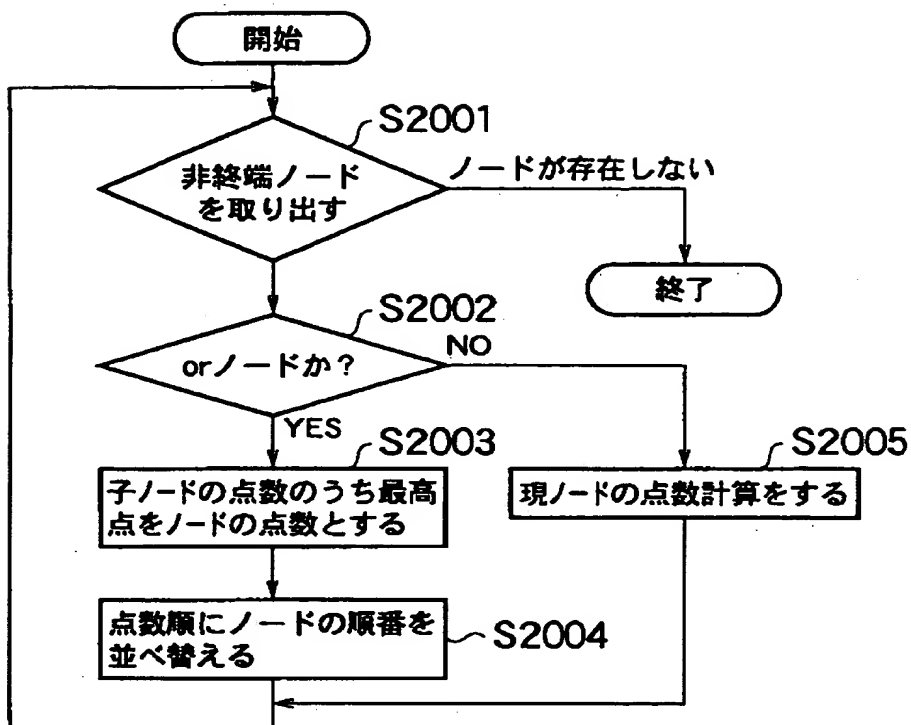
識別番号 [000000295]

1. 変更年月日	1990年 8月22日
[変更理由]	新規登録
住 所	東京都港区虎ノ門1丁目7番12号
氏 名	沖電気工業株式会社

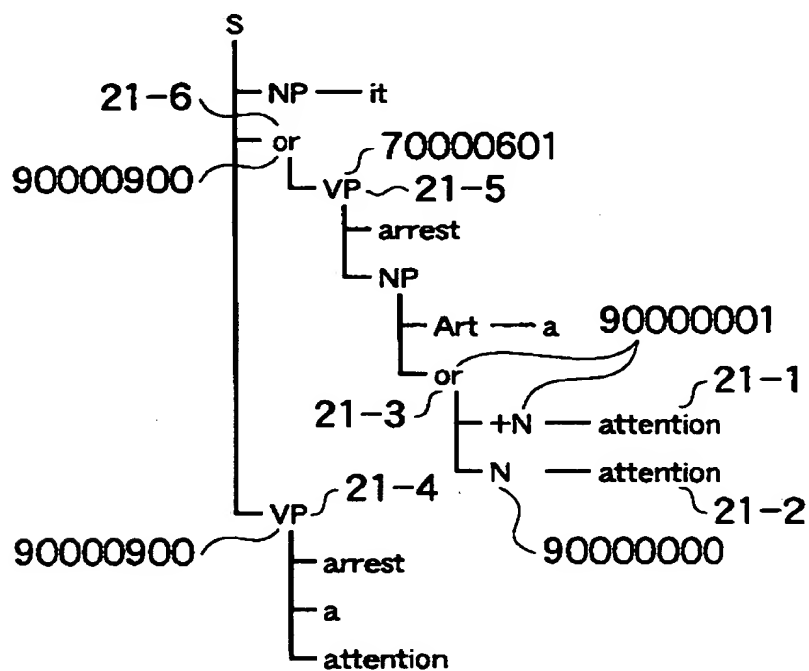
【図 1 9】



【図 20】



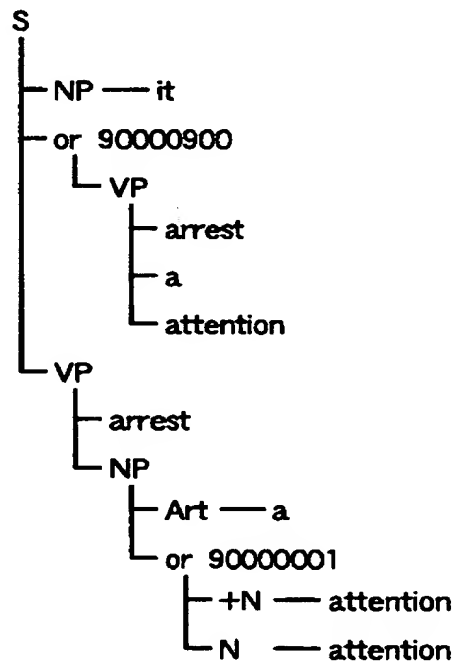
【図 21】



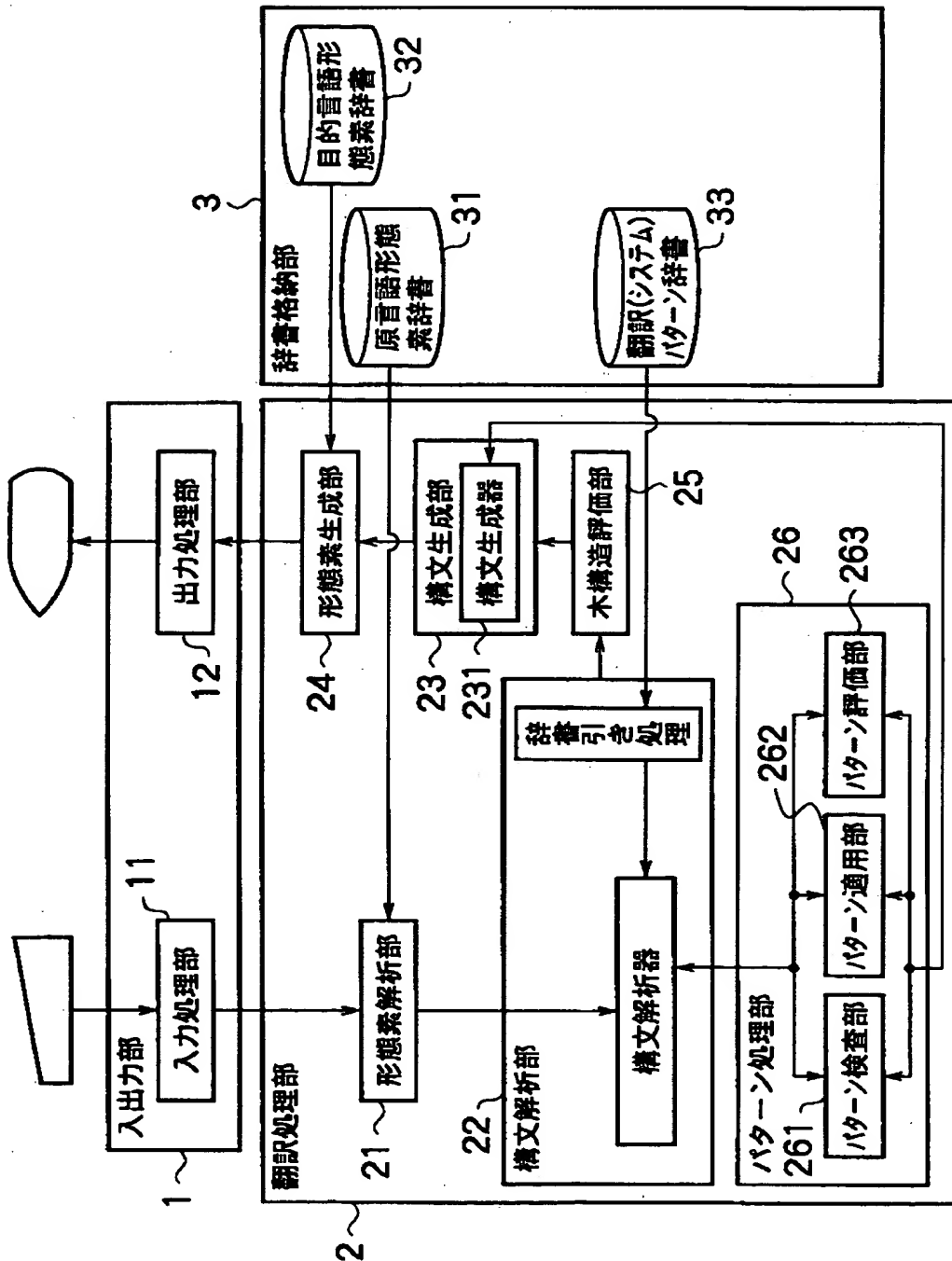
【図 2 2】

	221	222	223
	261a		
224	user	child	100000000
225	terminal	total	100000000-100000000N
226	!	child	1000000
227	*	total	100000-10000N
228	node	total	1000-100N
229	!	total	10N
230	+	total	1N

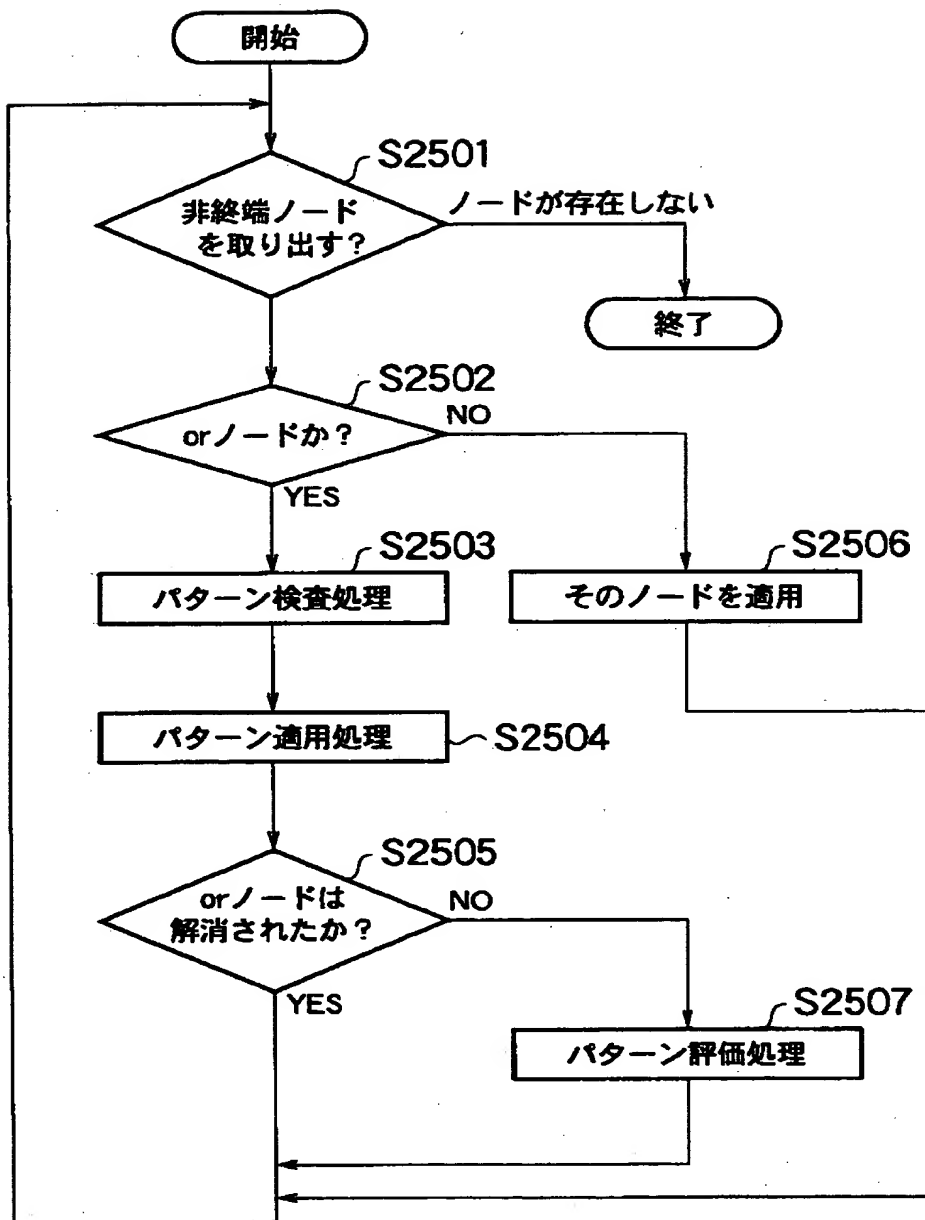
【図 2 3】



【図 24】

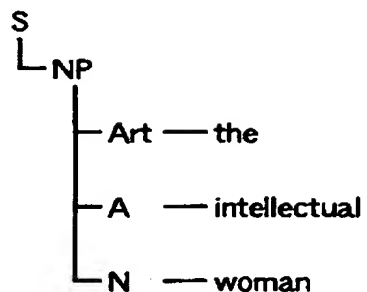


【図 25】

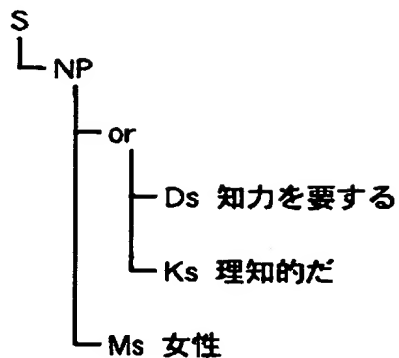


【図 2 6】

(a)

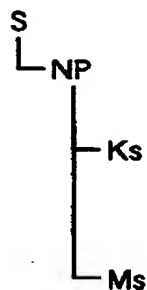


(b)



【図 2 7】

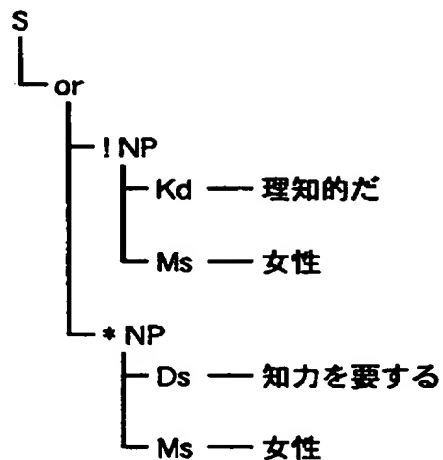
(a)



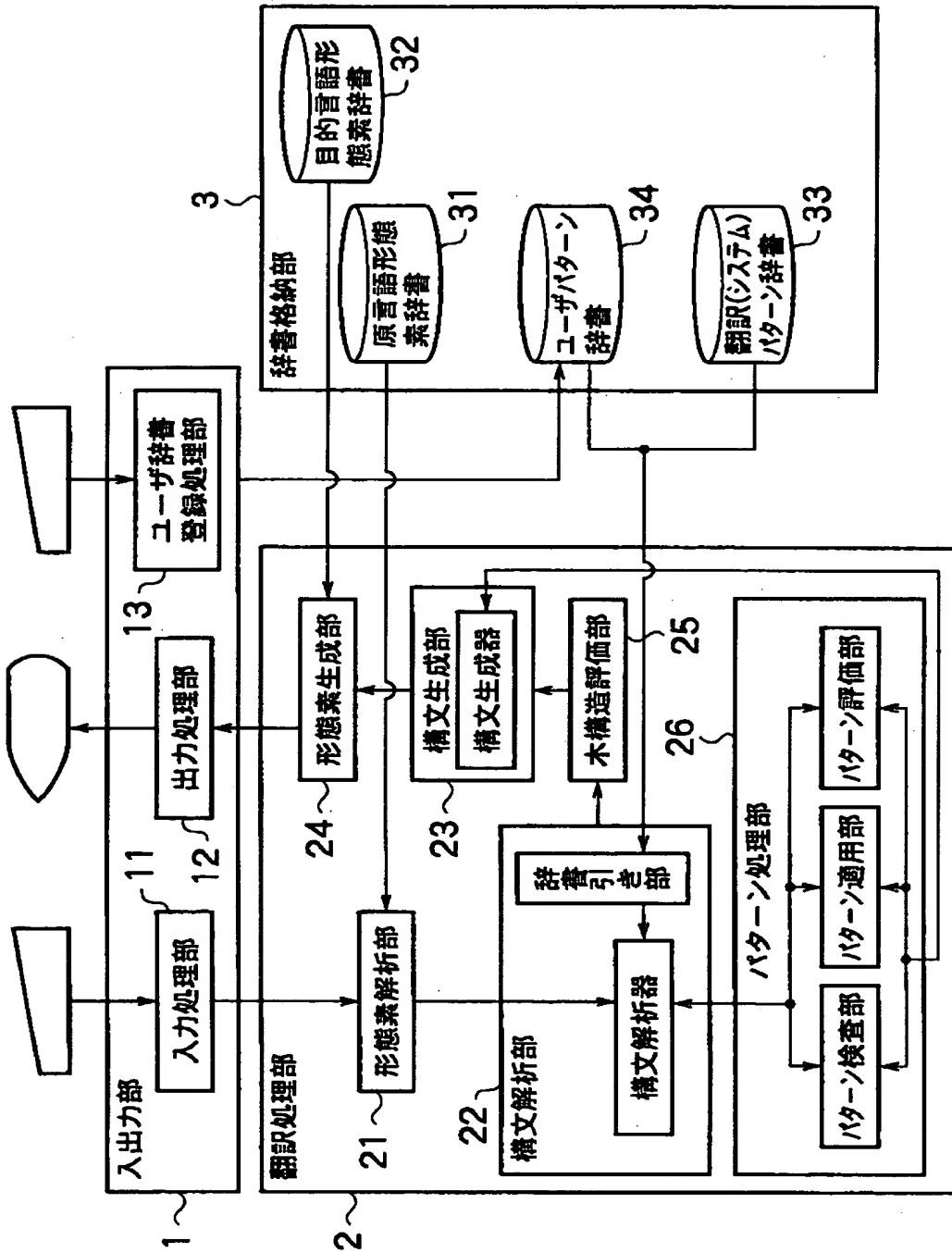
(b1) [en: NP [1: Art] [2: A] [3: N] ! [ja: NP [2: Ks] [3: Ms]]]

(b2) [en: NP [1: Art] [2: A] [3: N] \* [ja: NP [2: Ds] [3: Ms]]]

【図 2 8】



【図 29】





【書類名】 要約書

【要約】

【課題】 適切な自然言語処理結果（構文解析結果及び又は構文生成結果）を提供し得る自然言語処理装置を実現する。

【解決手段】 本発明は、少なくともパターン名及びパターン構成要素を有する自然言語パターンを利用して、構文解析、及び又は、構文生成を行う自然言語処理装置に関する。そして、予めパターン辞書に用意されている上記自然言語パターンから、構文解析、及び又は、構文生成時の処理での候補となる 1 以上の自然言語パターンを抽出する辞書引き手段と、候補の自然言語パターンが木構造に適合しているか否かを検査するパターン検査手段と、適合する場合に、その自然言語パターンを木構造に適用するパターン適用手段とを少なくとも有することを特徴とする。

【選択図】 図 1

【書類名】 手続補正書

【整理番号】 KN002336H

【提出日】 平成12年 7月28日

【あて先】 特許庁長官 及川 耕造 殿

【事件の表示】

【出願番号】 特願2000-225911

【補正をする者】

【識別番号】 000000295

【氏名又は名称】 沖電気工業株式会社

【代表者】 篠塚 勝正

【代理人】

【識別番号】 100090620

【弁理士】

【氏名又は名称】 工藤 宣幸

【手続補正 1】

【補正対象書類名】 特許願

【補正対象項目名】 発明者

【補正方法】 変更

【補正の内容】

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 北村 美穂子

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 村田 稔樹

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会

社内

【氏名】 佐々木 美樹

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 下畑 さより

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 福居 毅至

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 淵上 正睦

【その他】 第6番目の発明者の氏名は正しくは、淵上 正睦である  
所、願書作成時のワープロ入力で変換を誤って、淵上  
正睦と記載した。そのため、正しい氏名、淵上 正睦に  
訂正する

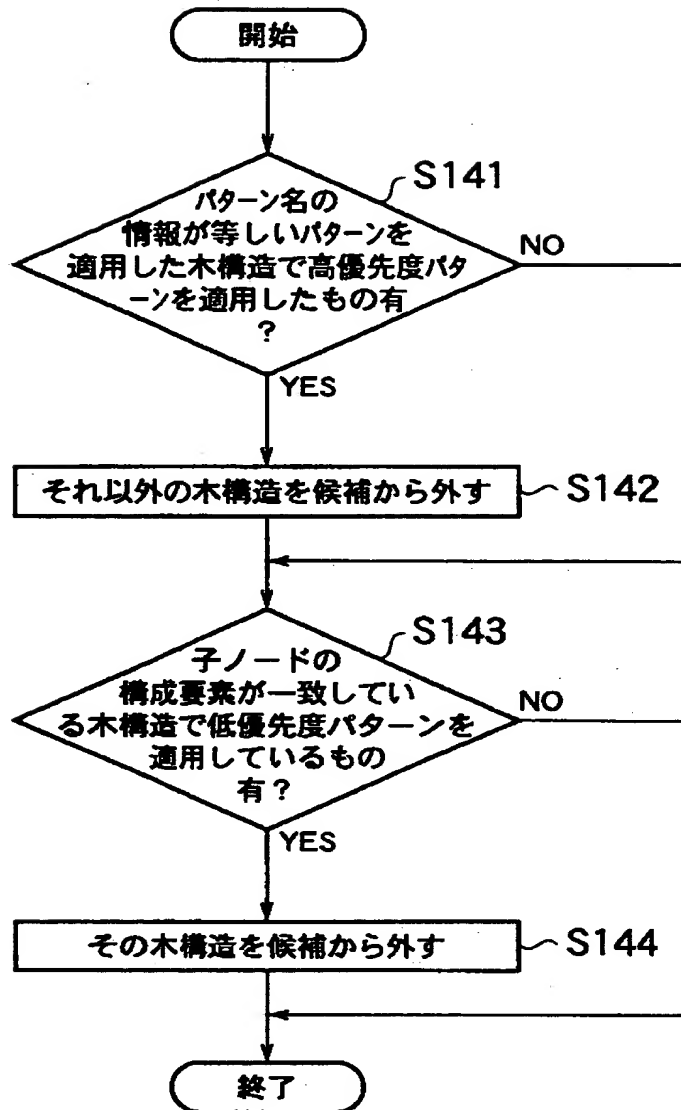
【ブルーフの要否】 要

出 願 人 履 歴 情 報

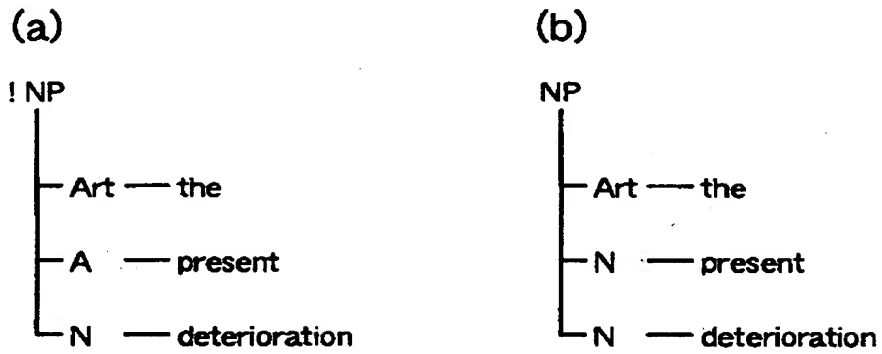
識別番号 [000000295]

1. 変更年月日	1990年 8月22日
[変更理由]	新規登録
住 所	東京都港区虎ノ門1丁目7番12号
氏 名	沖電気工業株式会社

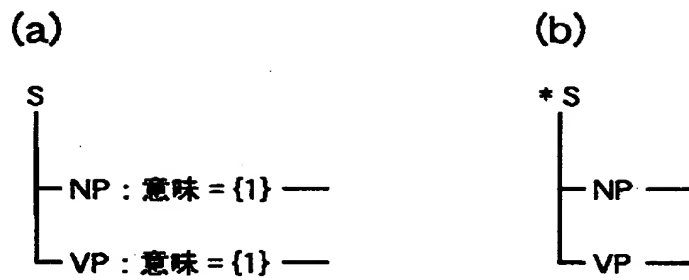
【図 14】



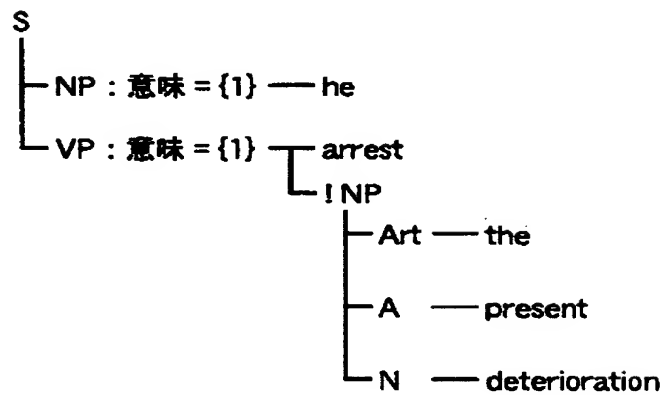
【図 15】



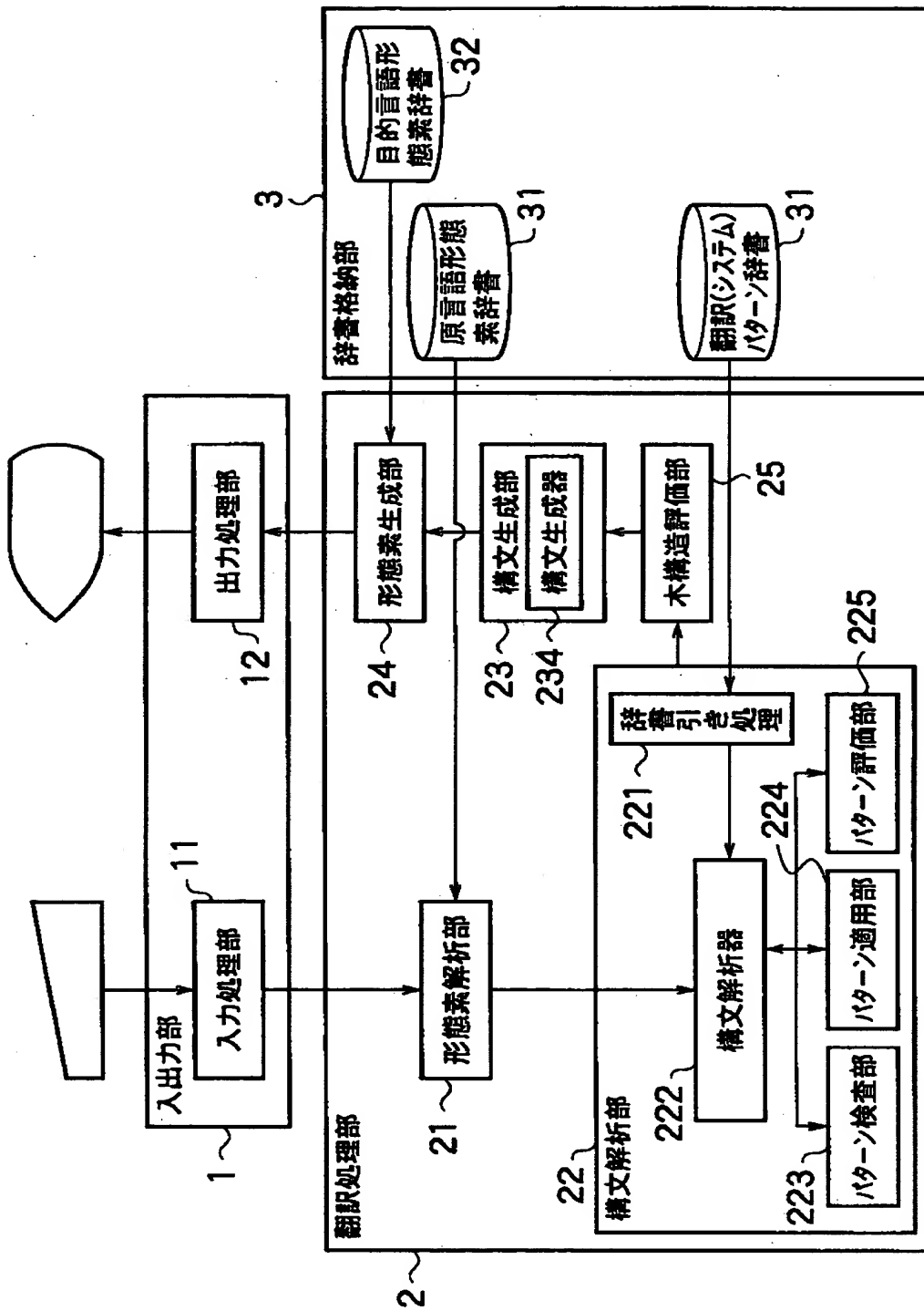
【図 16】



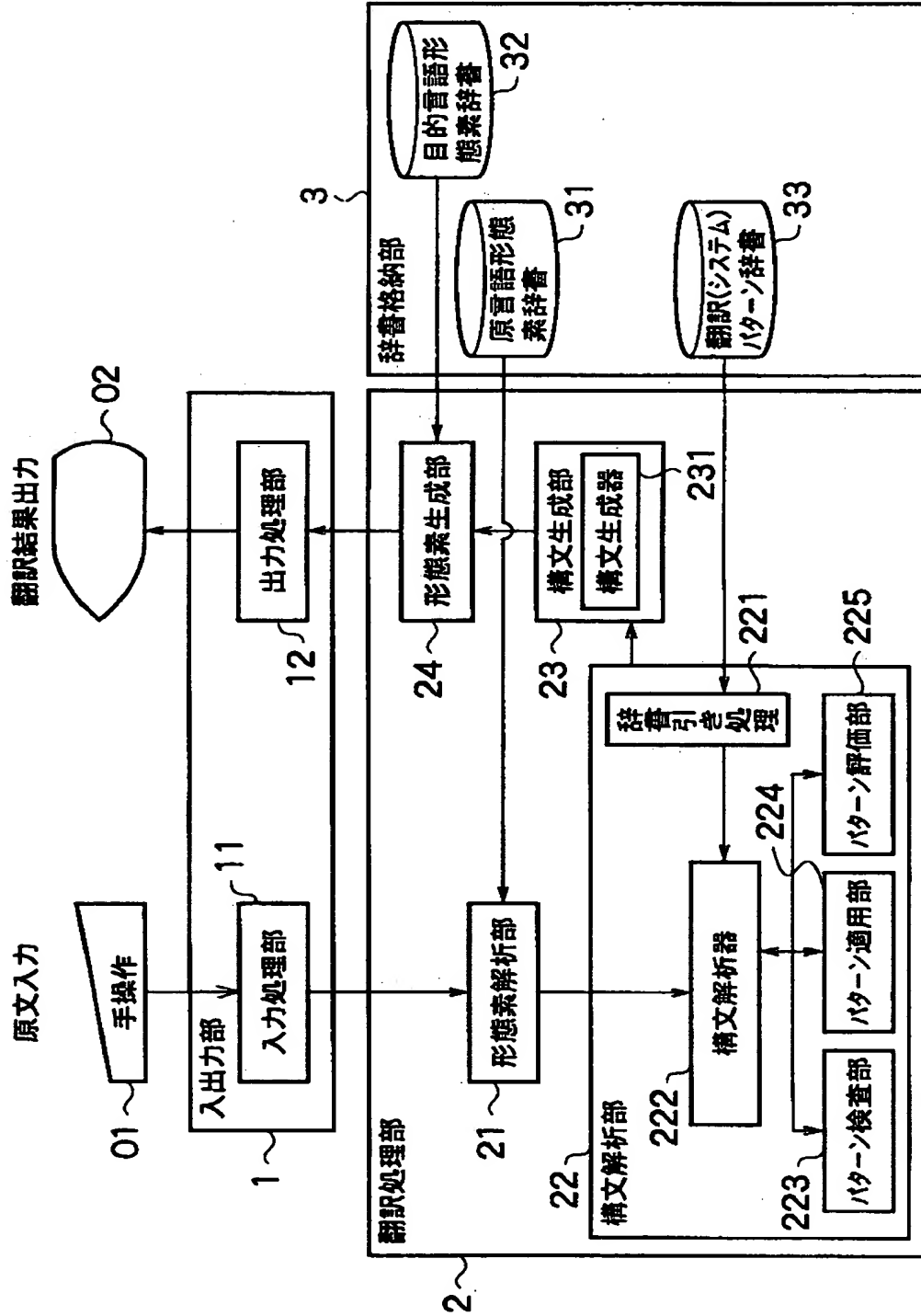
【図 17】



【図18】

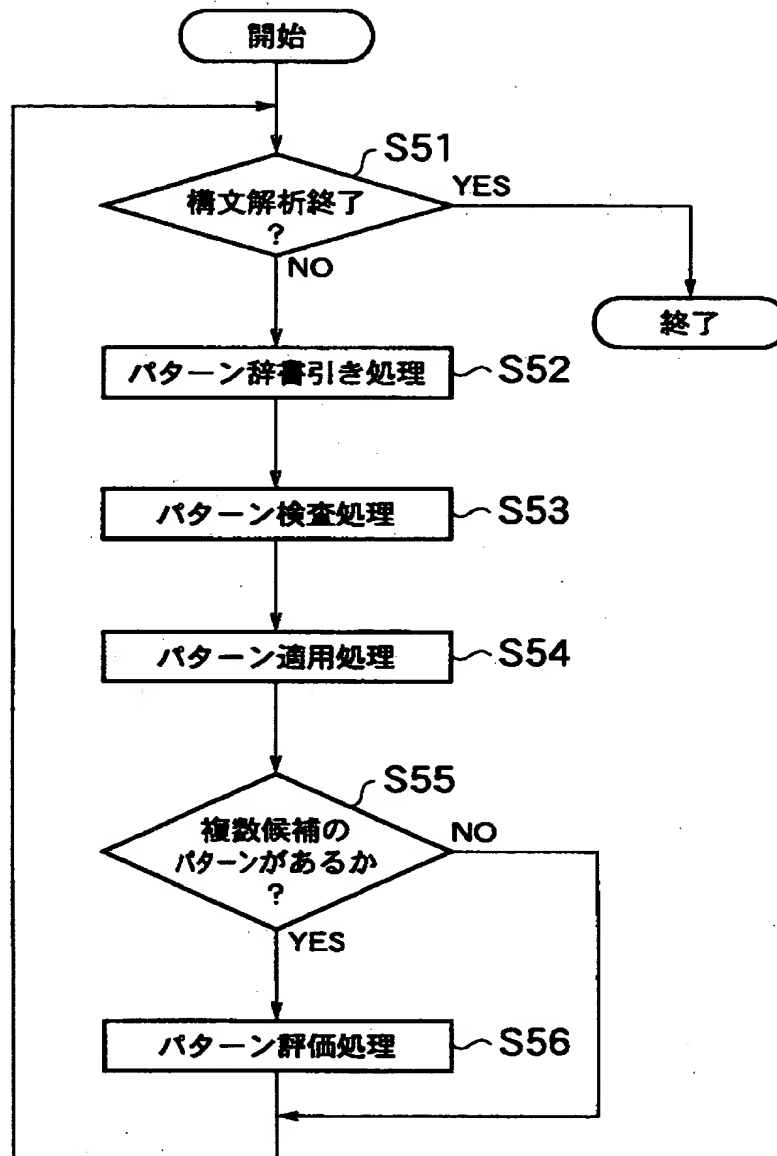


【図10】





【図11】



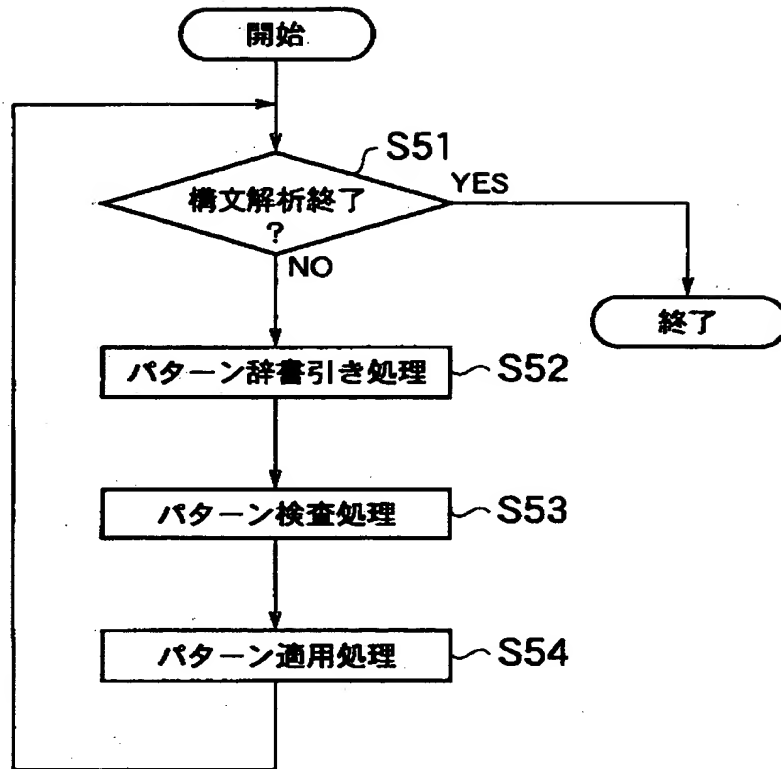
【図 1 2】

- 12-1 ~ [ en:VP arrest:品詞 = v [1:NP:意味 = 人]] [ ja:VP [1:NP] を逮捕する]
- 12-2 ~ [ en:VP arrest:品詞 = v [1:NP:意味 != 人]] [ ja:VP [1:NP] を阻止する]
- 12-3 ~ [ en:VP arrest:品詞 = v a attention] [ ja:VP 興味を引く]
- 12-4 ~ [ en:NP:意味 = 物 it:品詞 = prn] [ ja:NP it]
- 12-5 ~ [ en:Art the:品詞 = art] [ ja:Art]
- 12-6 ~ [ en:N:意味 = 状態 deterioration:品詞 = n] [ ja:Ms 悪化]
- 12-7 ~ [ en:N present 品詞 = n] [ ja:Ms 現在]
- 12-8 ~ [ en:A present:品詞 = adj] [ ja:Ks 現在の]
- 12-9 ~ [ en:NP:意味 = {1}[1:Art][2:A][3:N:意味 = {1}]] [ ja:NP [2:A][3:N]]
- 12-10 ~ [ en:NP:意味 = {1}[1:Art][2:N][3:N:意味 = {1}]] [ ja:NP [2:N][3:N]]
- 12-11 ~ [ en:S [1:NP:意味 = {意味}][2:VP:主格意味 = {意味}]] [ ja:S [1:NP] が [2:VP]]
- 12-12 ~ \* [ en:S [1:NP][2:VP]] [ ja:S [1:NP] が [2:VP]]

【図 1 3】



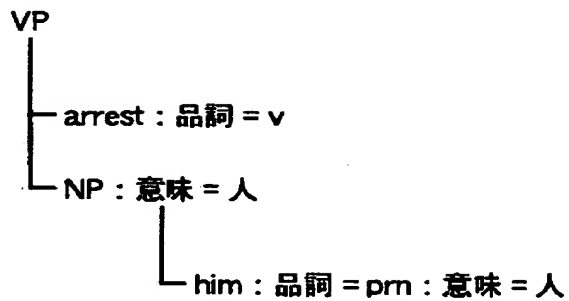
【図 5】



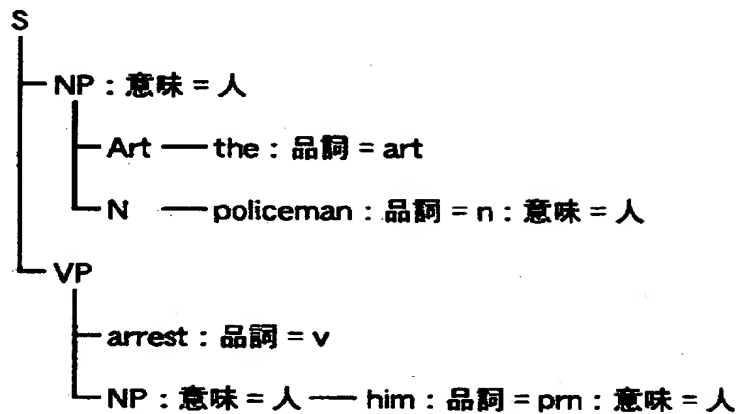
【図 6】



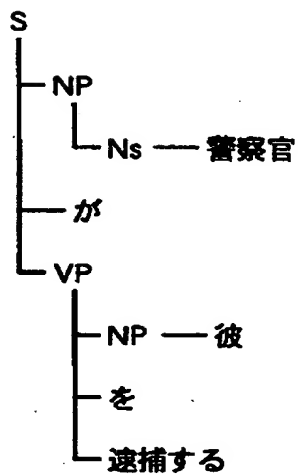
【図 7】



【図 8】



【図 9】



）である。

【図 1 6】

第 2 の実施形態のパターン評価処理の対象となる木構造候補を示す説明図（2）である。

【図 1 7】

第 2 の実施形態の構文解析結果の例を示す説明図である。

【図 1 8】

第 3 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【図 1 9】

第 3 の実施形態の機械翻訳装置の翻訳動作を示すフローチャートである。

【図 2 0】

第 3 の実施形態の木構造評価処理を示すフローチャートである。

【図 2 1】

第 3 の実施形態の木構造評価処理に供する構文解析結果例を示す説明図である。

【図 2 2】

第 3 の実施形態の木構造の評価点計算方法定義ファイルを示す説明図である。

【図 2 3】

第 3 の実施形態の木構造評価処理語の構文解析結果を示す説明図である。

【図 2 4】

第 4 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【図 2 5】

第 4 の実施形態の構文生成処理の特徴処理を示すフローチャートである。

【図 2 6】

第 4 の実施形態の構文生成処理の特徴処理を行う前の木構造を示す説明図である。

【図 2 7】

第 4 の実施形態のパターン検査処理及びパターン適用処理の説明図である。

【図 2 8】

第 4 の実施形態の構文生成結果例を示す説明図である。

【図 2 9】

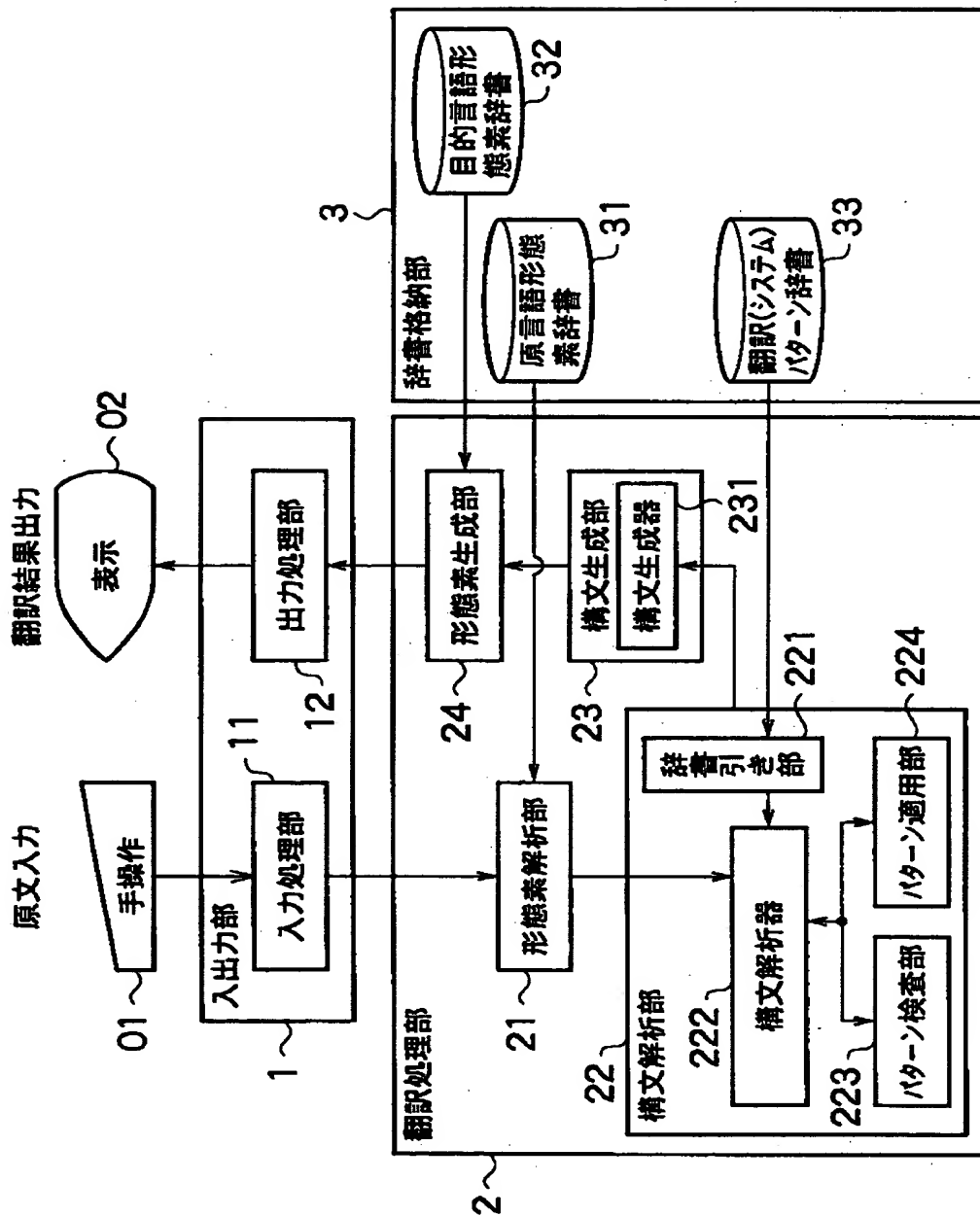
第 5 の実施形態の機械翻訳装置の機能的構成を示すブロック図である。

【符号の説明】

- 1 …入出力部、
- 2 …翻訳処理部、
- 3 …辞書格納部、
- 1 3 …ユーザ辞書登録処理部、
- 2 1 …形態素解析部、
- 2 2 …構文解析部、
- 2 3 …構文生成部、
- 2 4 …形態素生成部、
- 2 5 …木構造評価部、
- 2 6 …パターン評価部、
- 2 2 1 …辞書引き部、
- 2 2 3、2 6 1 …パターン検査部、
- 2 2 4、2 6 2 …パターン適用部、
- 2 2 5、2 6 3 …パターン評価部、
- 3 3 …翻訳パターン辞書（システムパターン辞書）、
- 3 4 …ユーザパターン辞書。

【書類名】 図面

【図 1】



【図2】

